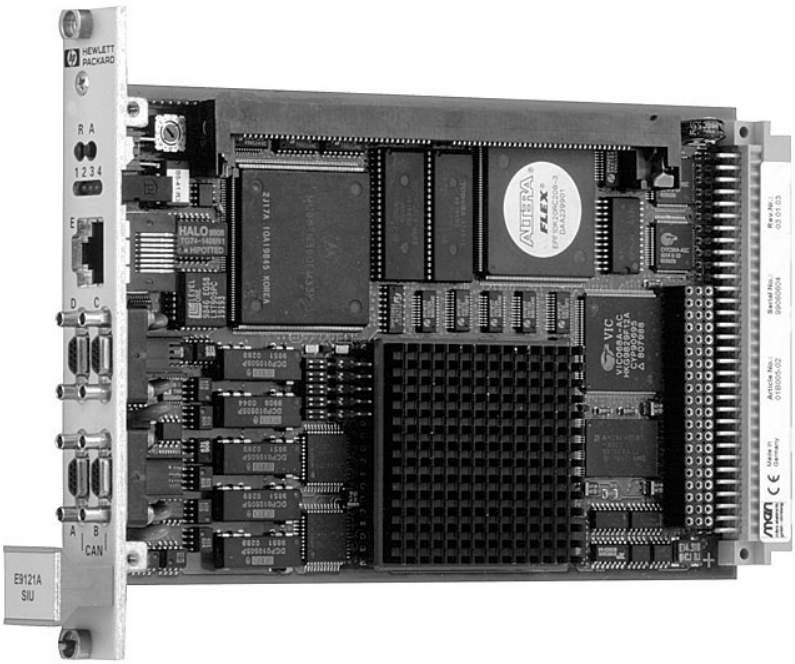
Embedded Solutions

B5 – 3U VMEbus High-Speed Communication Controller



User Manual

Board-Level Computers for Industrial Applications

men mikro elektronik gmbh • nürnberg

B5 – 3U VMEbus High-Speed Communication Controller

Due to its numerous serial interfaces and the powerful 68040V/68060 CPU, the B5 is optimized for high-speed communication control. All six serial lines of the 68360 are available: one Ethernet 10Base-T, two RS232 interfaces, two RS232, RS422, RS485 or TTY interfaces via SA adapters (or optional second Ethernet) and one ISO 9141-2 automotive interface. Two full/extended CAN interfaces complete the B5's communication features.

The B5 supports full VMEbus master/slave functionality; it has abundant on-board memory and an RTC.

Due to its local and global bus, genuine dual-ported DRAM and the shared M-Module slot the B5 is also optimized for slave operation with all VME communication features installed.

The B5 is able to run without VMEbus, being a real single-board computer with an M-Module slot for flexible I/ O extension.

The B5 is an ideal solution for automotive applications, especially for small mobile systems and harsh environmental conditions.

Technical Data

CPU

- MC68040/33MHz (or option: MC68060/50MHz)

Peripheral Controller

- 32-bit CPU MC68360/33MHz

VMEbus

- 3U VMEbus form factor
- VMEbus master/slave interface
  - VIC068
  - Interrupter/interrupt handler
  - A16, A24, D16
  - VMEbus slot-1 functionality

Memory

- Up to 512KB boot Flash
  - 16-bit data bus
  - Dual-ported

- Up to 1MB SRAM
  - Battery-backed via VMEbus
  - 16-bit data bus
  - Dual-ported
- Up to 4MB DRAM on board
  - 32-bit data bus
  - Burst access
- Up to 32 MB DRAM
  - 1 JEDEC SIMM module
  - 32-bit data bus
  - Burst access
- Up to 8MB Flash
  - 32-bit data bus
  - Onboard programming
  - Burst access

Interfaces

- 2 SMC UARTs
  - RS232 interfaces
  - Optically isolated
  - 9-pin micro D-Sub connector at front panel
- 1 SCC intelligent serial interface
  - ISO 9141-2 interface, option: RS232
  - Optically isolated
  - 9-pin micro D-Sub connector at front panel
- Ethernet controller
  - CPM of the MC68360
  - Local DMA
  - 10Base-T using RJ45 connector
  - 10 Mbits/s data transfer rate
- 2 serial SA adapter interfaces
  - Physical interface using adapter (RS232..RS485, optically isolated or not) on 10-pin ribbon-cable connector
  - One interface also useable for Ethernet (adapter in preparation)
- CAN bus interfaces
  - 2 full CAN controllers i82527 with extended addressing
  - ISO 11898-High Speed up to 1Mbit/s, both channels optically isolated using DC/DC converters
  - Two 9-pin micro D-Sub connectors at front panel

M-Module Extension

- 1 M-Module slot (requires additional VMEbus slot and front panel)
- Characteristics: A08, A24, D16, INTA, INTC, TRIGA, TRIGB, DMA
- D32, DMA, TRIGA, TRIGB access is supported only for local CPU
- Dual-ported

Miscellaneous

- Battery-backed real-time clock
- Programmable watchdog
- Serial EEPROM (4Kbit) for setup
- Hex switch
- Four programmable LEDs
- Reset button and abort button
- Single 5V supply (12V for M-Modules if installed)

Electrical Specifications

- Optical isolation: 1kV DC
- Isolation voltage for Ethernet/CAN bus: 500V DC
- Supply voltage/power consumption: +5V (4.85V..5.25V), 1.5A typ.
- MTBF: 45,000h @ 50°C

Mechanical Specifications

- Dimensions: standard single Eurocard, 100mm x 160mm
- Weight: 260g

Environmental Specifications

- Temperature range (operation):
  - 0..+60°C or -40..+85°C
  - Airflow: min. 10m³/h
- Temperature range (storage): -40..+85°C
- Relative humidity range (operation): max. 95% non-condensing
- Relative humidity range (storage): max. 95% non-condensing
- Altitude: -300m to + 3,000m
- Shock: 15g/0.33ms, 6g/6ms
- Vibration: 1g/5..2,000Hz

Safety

- PCB manufactured with a flammability rating of 94V-0 by UL recognized manufacturers

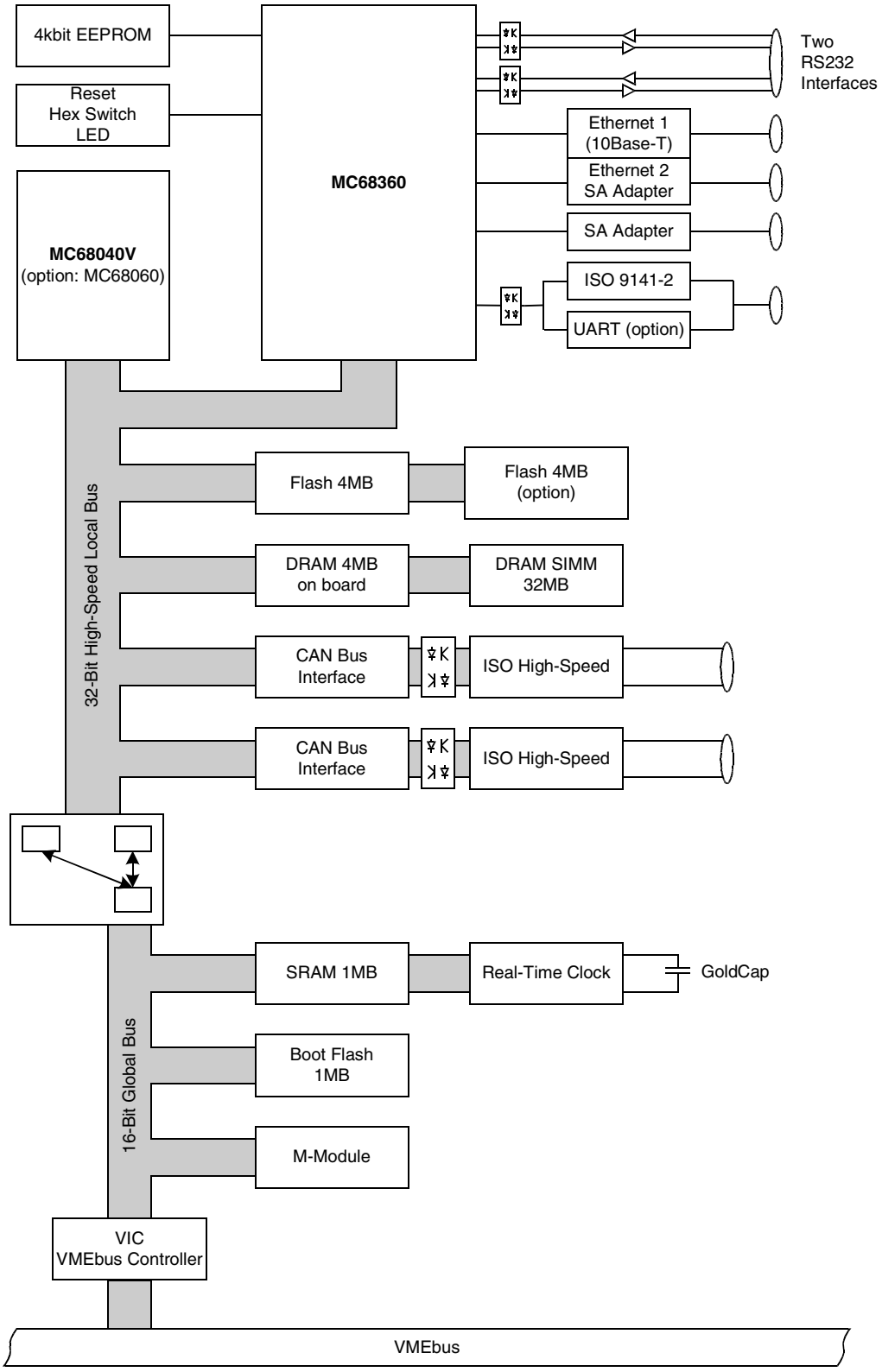
EMC

- Tested according to EN 55022 / 1999-05 (radio disturbance) and EN 55024 / 1999-05 (immunity) with regard to CE conformity

Software Support

- VxWorks
- OS-9
- MENMON

Block Diagram



Product Safety

Fuses

This board contains fuses. If you need to replace a fuse, make sure you adhere to the following types and ratings:

Component	Current Rating	Type	Size
S1	1.5A	Fast	1206
S2	1.5A	Fast	1206
S3	1.5A	Fast	1206
S4	1.5A	Fast	1206

For component locations, see Chapter 5.3 Component Plans on page 66.

Electrostatic Discharge (ESD)

Computer boards and components contain electrostatic sensitive devices. Electrostatic discharge (ESD) can damage components. To protect the board and other components against damage from static electricity, you should follow some precautions whenever you work on your computer.

- Power down and unplug your computer system when working on the inside.
- Hold components by the edges and try not to touch the IC chips, leads, or circuitry.
- Use a grounded wrist strap before handling computer components.
- Place components on a grounded antistatic pad or on the bag that came with the component whenever the components are separated from the system.
- Store the board only in its original ESD-protected packaging. Retain the original packaging in case you need to return the board to MEN for repair.

About this Document

This user manual describes the hardware functions of the board, connection of peripheral devices and integration into a system. It also provides additional information for special applications and configurations of the board.

The manual does not include detailed information on individual components (data sheets etc.). A list of literature is given in the appendix.

History

Edition	Description	Technical Content	Date of Issue
E1	First edition	Jürgen Steinert, Klaus Popp	1998-02-13
E2	Second edition	Manfred Schmitz, Klaus Popp	2003-01-15

Conventions

This sign marks important notes or warnings concerning proper functionality of the product described in this document. You should read them in any case.

italics — Folder and file names are printed in *italics*.

bold — **Bold** type is used for emphasis.

hyperlink — Hyperlinks are printed in blue color.

The globe will show you where hyperlinks lead directly to the Internet, so you can look for the latest information online.

`0xFF` — Hexadecimal numbers are preceded by "0x", which is the usual C-language convention, and are printed in a monospace type, e.g. `0x00FFFF`.

IRQ# /IRQ — Signal names followed by "#" or preceded by a slash ("/") indicate that this signal is either active low or that it becomes active at a falling edge.

in/out — Signal directions in signal mnemonics tables generally refer to the corresponding board or component, "in" meaning "to the board or component", "out" meaning "coming from it".

Vertical lines on the outer margin signal technical changes to the previous edition of the document.

Copyright Information

MEN reserves the right to make changes without further notice to any products herein. MEN makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does MEN assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages.
"Typical" parameters can and do vary in different applications. All operating parameters, including "Typicals" must be validated for each customer application by customer's technical experts.
MEN does not convey any license under its patent rights nor the rights of others.
MEN products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the MEN product could create a situation where personal injury or death may occur. Should Buyer purchase or use MEN products for any such unintended or unauthorized application, Buyer shall indemnify and hold MEN and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that MEN was negligent regarding the design or manufacture of the part.

All brand or product names are trademarks or registered trademarks of their respective holders.

Information in this document has been carefully checked and is believed to be accurate as of the date of publication; however, no responsibility is assumed for inaccuracies. MEN will not be liable for any consequential or incidental damages arising from reliance on the accuracy of this document. The information contained herein is subject to change without notice.

Copyright © 2003 MEN Mikro Elektronik GmbH. All rights reserved.

Please recycle

Germany
MEN Mikro Elektronik GmbH
Neuwieder Straße 7
90411 Nuremberg
Phone +49-911-99 33 5-0
Fax +49-911-99 33 5-99
E-mail info@men.de
www.men.de

France
MEN Mikro Elektronik SA
18, rue René Cassin
ZA de la Châtelaine
74240 Gaillard
Phone +33 (0) 450-955-312
Fax +33 (0) 450-955-211
E-mail info@men-france.fr
www.men-france.fr

UK
MEN Micro Ltd
Whitehall, 75 School Lane
Hartford, Northwich
Cheshire UK, CW8 1PF
Phone +44 (0) 1477-549-185
Fax +44 (0) 1477-549-178
E-mail info@menmicro.co.uk
www.menmicro.co.uk

USA
MEN Micro, Inc.
3740 North Josey Lane, Suite 203
Carrollton, TX 75007
Phone 972-939-2675
Fax 972-939-0055
E-mail sales@menmicro.com
www.menmicro.com

Contents

1 Getting Started 13
1.1 Map of the Board 13
1.2 Configuring the Hardware 14
1.3 Integrating the Board into a System 15
1.4 Installing an M-Module and SA Adapter 16
1.4.1 B5 Accessory Kit 16
1.4.2 Installation 16
1.5 Installing Operating System Software 18

2 Functional Description 19
2.1 Power Supply 19
2.1.1 Connection of Supply Voltage 19
2.1.2 SRAM and Real-Time Clock Battery 19
2.2 Bus Structure 20
2.3 Processor Core 21
2.4 Memory 22
2.4.1 Boot Flash (Global Bus) 22
2.4.2 Static RAM (Global Bus) 23
2.4.3 Flash (Local Bus) 23
2.4.4 Dynamic RAM (Local Bus) 24
2.4.5 Serial EEPROM 25
2.5 Ethernet Interface 26
2.5.1 Connection 26
2.5.2 General 26
2.5.3 10Base-T 27
2.6 CAN Bus Interfaces (Local Bus) 28
2.6.1 General 28
2.6.2 Basic CAN, Full CAN and Extended CAN 29
2.6.3 Implementation on the Board 29
2.6.4 Connection 30
2.7 Serial Interfaces 31
2.7.1 RS232 Interfaces 31
2.7.2 ISO 9141-2/RS232 Interface 32
2.7.3 Optional Serial Interfaces 33
2.8 M-Module Slot (Global Bus) 35
2.8.1 Connection 35
2.8.2 Addressing the M-Module 36
2.8.3 DMA 36
2.8.4 Interrupts 37
2.9 VMEbus Interface 38
2.9.1 Slot-1 Function 38
2.9.2 VMEbus Master Interface 38
2.9.3 VMEbus Slave Interface 39
2.9.4 VMEbus Interrupt Handler 40

2.9.5 Utility Bus Signals ACFAIL# and SYSFAIL# 40
2.9.6 VMEbus Interrupter 40
2.9.7 Remote2 Line 40
2.9.8 Stand-Alone Operation 40
2.9.9 Connection 41
2.10 Real-Time Clock 42
2.11 User LEDs 42
2.12 Reset/Abort Button 42
2.13 Watchdog Logic 42
2.14 On-Board Bus Error Logic 42

3 The MENMON Debugger 43
3.1 General 43
3.2 Console 44
3.2.1 Serial Interface 44
3.3 B5 MENMON Memory Map 44
3.4 MENMON Start-Up 45
3.4.1 DRAM/SRAM Recognition 45
3.5 Updating Flash/Serial EEPROM 46
3.5.1 Update File 46
3.5.2 Update via a Serial Interface 47
3.5.3 Update via VMEbus 47
3.6 MENMON User Interface 48
3.6.1 Command Line Editing 48
3.6.2 Numerical Arguments 48
3.6.3 Commands 49
3.7 Exception Handling 53
3.8 I/O Branch Table 53

4 Organization of the Board 55
4.1 Address Mappings 55
4.1.1 Local and Global Bus Access by Local CPU 55
4.1.2 Global Bus Access by Another VMEbus Master 56
4.2 Interrupt Handling 57
4.3 B5 Control Registers 58
4.3.1 Local Control Registers 58
4.3.2 Global Control Registers 59
4.4 Implementation of MC68360 61

5 Appendix 64
5.1 Literature and WWW Resources 64
5.1.1 CPU 64
5.1.2 Ethernet 64
5.1.3 CAN Bus 64
5.1.4 VMEbus 65
5.1.5 M-Modules 65
5.2 Board Revisions 65
5.3 Component Plans 66

Figures

Figure 1. Map of the Board - Front Panel and Top View 13
Figure 2. Hex Switch for first Start-up 15
Figure 3. Screws for Fastening of M-Module 16
Figure 4. Alignment of Mounting Board and B5 17
Figure 5. Bus Structure 20
Figure 6. DRAM Installation 24
Figure 7. Byte Ordering for CAN Controllers 29
Figure 8. Position of P9 and P10 33
Figure 9. Hex Switch for VMEbus A16 Slave Address 39
Figure 10. B5-Specific Branch Table Structure 53
Figure 11. Component Plan of B5 Hardware Revision 04 — top side 66
Figure 12. Component Plan of B5 Hardware Revision 04 — bottom side 66

Tables

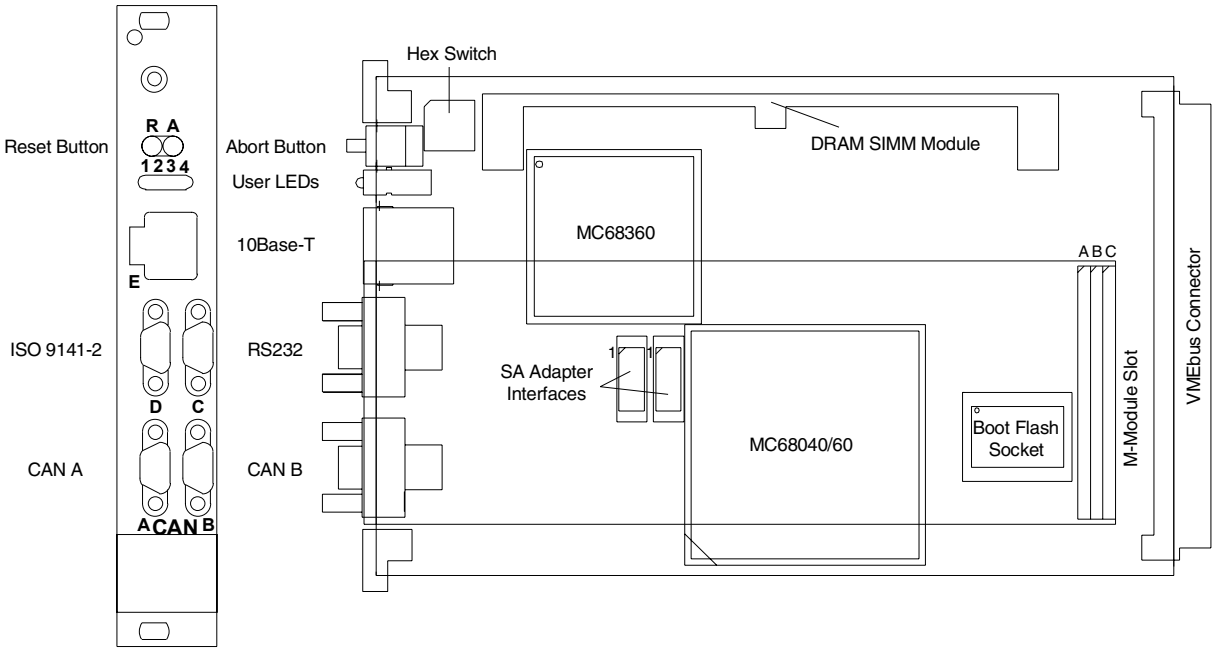
Table 1. Terminal Lines of the 9-pin micro D-Sub RS232 Plug Connector (P2) 15
Table 2. Power Supply 19
Table 3. Compare Chart for MC68360, MC68040V and MC68060 21
Table 4. Pin Assignment of the 8-pin RJ45 Ethernet 10Base-T Connector 26
Table 5. Signal Mnemonics of the Ethernet 10Base-T Connector 26
Table 6. CAN Bus Transfer Rates related to Line Lengths and Cables 28
Table 7. Pin Assignment of the 9-pin micro D-Sub CAN A Plug Connector . . 30
Table 8. Pin Assignment of the 9-pin micro D-Sub CAN B Plug Connector . . 30
Table 9. Signal Mnemonics of the CAN Bus Connectors 30
Table 10. Pin Assignment of the 9-pin micro D-Sub RS232 Plug Connector . . . 31
Table 11. Signal Mnemonics of the RS232 Interfaces 31
Table 12. Pin Assignment of the 9-pin micro D-Sub ISO 9141-2 Plug Connector 32
Table 13. Signal Mnemonics of the ISO 9141-2 Interface 32
Table 14. Pin Assignment of the 10-Pin Plug Connector P9 34
Table 15. Pin Assignment of the 10-Pin Plug Connector P10 34
Table 16. Signal Mnemonics of the Optional Serial Interfaces 34
Table 17. Pin Assignment of the 60-Pin M-Module Receptacle Connector 35
Table 18. VMEbus Interrupt Levels 40
Table 19. Pin Assignment of the 3-Row, 96-Pin VMEbus Connector P1 41
Table 20. MENMON Memory Map 44
Table 21. Flash Sector Sizes 46
Table 22. Address Map for Local and Global Bus Access by Local CPU 55
Table 23. Address Map for Global Bus Access by A16 VMEbus Master 56
Table 24. Address Map for Global Bus Access by A24 VMEbus Master 56
Table 25. Interrupt Priorities and Levels 57
Table 26. Chip-Select Signals of MC68360 61
Table 27. Use of the MC68360 CPM 61
Table 28. Port Pin Assignment of the MC68360 CPM 62
Table 29. Table of Hardware Revisions 65

1 Getting Started

This chapter will give an overview of the B5 and some hints for first installation in a VMEbus system as a "check list".

1.1 Map of the Board

Figure 1. *Map of the Board - Front Panel and Top View*



1.2 Configuring the Hardware

You should check your hardware requirements before installing the board in a system, since most modifications are difficult or even impossible to do when the board is mounted in an enclosure.

The following check list will give an overview on what you might want to configure.

☑ DRAM SIMM modules

The B5 may be shipped without any DRAM on board, depending on the model ordered. You should check your main memory needs and install a suitable PS/2 SIMM module.

Refer to Chapter 2.4.4 Dynamic RAM (Local Bus) on page 24 for a detailed installation description and hints on supported SIMM modules.

☑ M-Module and SA adapters

Refer to Chapter 1.4 Installing an M-Module and SA Adapter on page 16 for a detailed installation description. Also, observe the installation recommendations given in the M-Module's user manual.

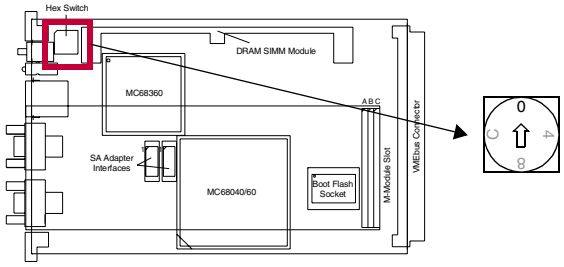
1.3 Integrating the Board into a System

The B5 is a complex board and setting it up requires experience. You can use the following check list when installing the CPU board in a VMEbus system for the first time and with minimum configuration.

The board is completely trimmed on delivery. Perform the following procedure without the M-Module installed!

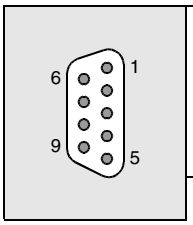
- ☑ Power-down the system.
- ☑ Remove all boards from the VMEbus system.
- ☑ Make sure the hex switch of the B5 is at position 0.

Figure 2. *Hex Switch for first Start-up*



- ☑ Install the B5 in slot 1 of the system.
- ☑ Connect a terminal to the standard RS232 interface (9-pin micro D-Sub connector) by wiring the following lines to the connector:

Table 1. *Terminal Lines of the 9-pin micro D-Sub RS232 Plug Connector (P2)*



6	-	1	-
7	-	2	RxD5
8	-	3	TxD5
9	-	4	-
		5	GND5

Note: The RS232 interface is not PC-compatible. Do **not** connect pins 1 and 4!

- ☑ Set your terminal to the following protocol:
  - 9600 baud data transmission rate
  - 8 data bits
  - 1 stop bit
  - no parity
- ☑ Switch on the system.
- ☑ The terminal displays the following message:

```
====[Secondary MENMON for B5 Version x.y]====
```

- ☑ Press **<CTRL> <A>** to enter MENMON.
- ☑ Now you can use the MENMON debugger (see detailed description in Chapter 3 The MENMON Debugger on page 43).
- ☑ Observe the installation instructions for the respective software.

1.4 Installing an M-Module and SA Adapter

1.4.1 B5 Accessory Kit

To install an M-Module and/or SA adapter you need MEN's B5 accessory kit (see MEN's website), which contains

- one 1-slot mounting board with an extra front panel for one M-Module and one SA adapter
- one cover for the M-Module slot
- one cover for the SA adapter slot
- one 60-pin contact strip for the M-Module
- one ribbon cable for the SA adapter
- fastening material.

1.4.2 Installation

Note: MEN gives no warranty on functionality and reliability of the B5, M-Modules and SA adapters used if you install M-Modules and/or SA adapters in a different way than described in this manual.

Perform the following steps to install an M-Module and/or an SA adapter:

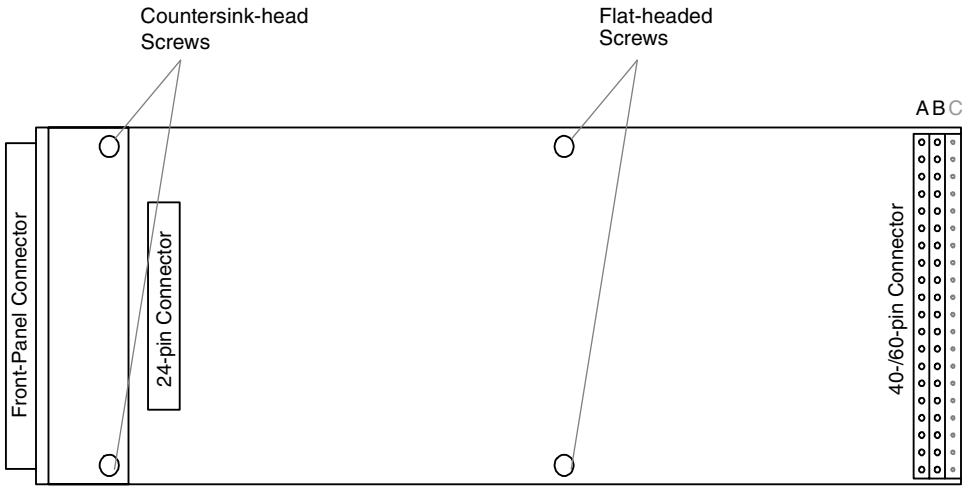
☑ Power-down your system and remove the B6 from the system.

M-Module

☑ Install the M-Module on the mounting board with its component side facing the board. The front connector extends through the front panel of the mounting board.

☑ Screw the M-Module to the mounting board using two countersink-head screws and two flat-headed screws from the accessory kit (see figure below).

Figure 3. *Screws for Fastening of M-Module*



SA Adapter

- ☑ Remove the two hexagonal-head bolts at the front of the SA-adapter connector.
- ☑ Plug one of the ribbon cable connectors to the SA-adapter's 10-pin plug connector. Make sure that you match the pins correctly (cf. Figure 1, Map of the Board - Front Panel and Top View, on page 13)!
- ☑ Place the SA adapter on the mounting board with its component side facing the board. The front connector extends through the front panel of the mounting board.
- ☑ Use the two hexagonal-head bolts removed from the front connector to screw the SA adapter tightly to the front panel.

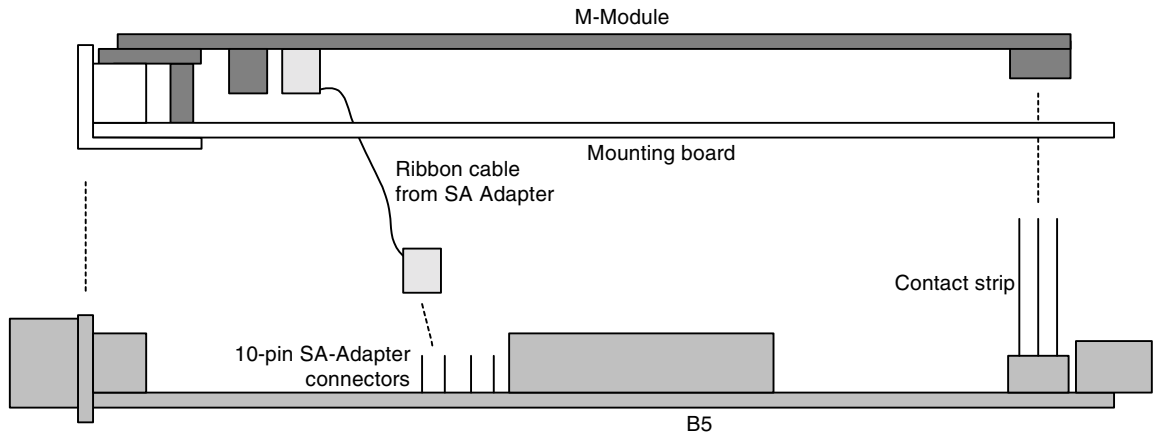
Plugging the Mounting Board on B5

- ☑ Plug the 60-pin contact strip from the accessory kit carefully but firmly to the M-Module connector of the B5.
- ☑ Hold the mounting board parallelly over the B5 as shown in the figure below. The ribbon cable of the SA adapter goes through the hole provided in the mounting board.
- ☑ Plug the 10-pin connector at the loose end of the SA-adapter ribbon cable to the B5. Again, make sure that you match the pins correctly.
  Be careful not to damage the 60-pin contact strip!
- ☑ Align the 40-/60-pin connector of the M-Module with the 60-pin contact strip.

Note: Older M-Modules often have only 40-pin connectors. They are plugged to rows A and B of the 60-pin connector of the B5 (cf. Figure 1, Map of the Board - Front Panel and Top View, on page 13).

- ☑ Press the contacts and M-Module connector carefully but firmly together.
- ☑ Now insert the two boards into the enclosure.

Figure 4. *Alignment of Mounting Board and B5*



1.5 Installing Operating System Software

The B5 fully supports OS-9 version 3.0.

By standard, no operating system is installed on the board. Please refer to MEN's OS-9 installation manual on how to install the software!

You can find any driver software available on MEN's website.

2 Functional Description

The following describes the individual functions of the B5 and their configuration on the board. There is no detailed description of the individual controller chips and the CPUs. They can be obtained from the data sheets or data books of the semiconductor manufacturer concerned (Chapter 5.1 Literature and WWW Resources on page 64).

2.1 Power Supply

2.1.1 Connection of Supply Voltage

The B5 is supplied with +5V and +12V via the VMEbus.

Table 2. *Power Supply*

U	I_{max}	I_{typ}	Function
+5V	2A	1.5A	Entire logic of the board Interface adapters for additional serial interfaces
±12V	1A	-	M-Module

If a CPU with 3.3V is used on B5, this voltage is gained from the 5V using a linear regulator.

2.1.2 SRAM and Real-Time Clock Battery

For uninterrupted power supply to the static RAM and real-time clock on the B5, an external battery must be connected (minimum 3V). Thus, the user himself is responsible for choosing a battery that represents the optimum solution for the respective application in terms of environmental protection, availability etc. Nominal battery voltage must be between 3V and 5V. The current necessary for data retention heavily depends on the quality of the SRAMs and ambient temperature. It is a typical 10µA for the RAMs. For instance, with a battery capacity of 1Ah this is sufficient for 10 years, if self-discharge of the battery is neglected.

When the battery is being changed, data retention is guaranteed for approx. three days if the battery voltage has not fallen below 2.6V or if the 5V supply has been switched on for a short time before changing the battery.

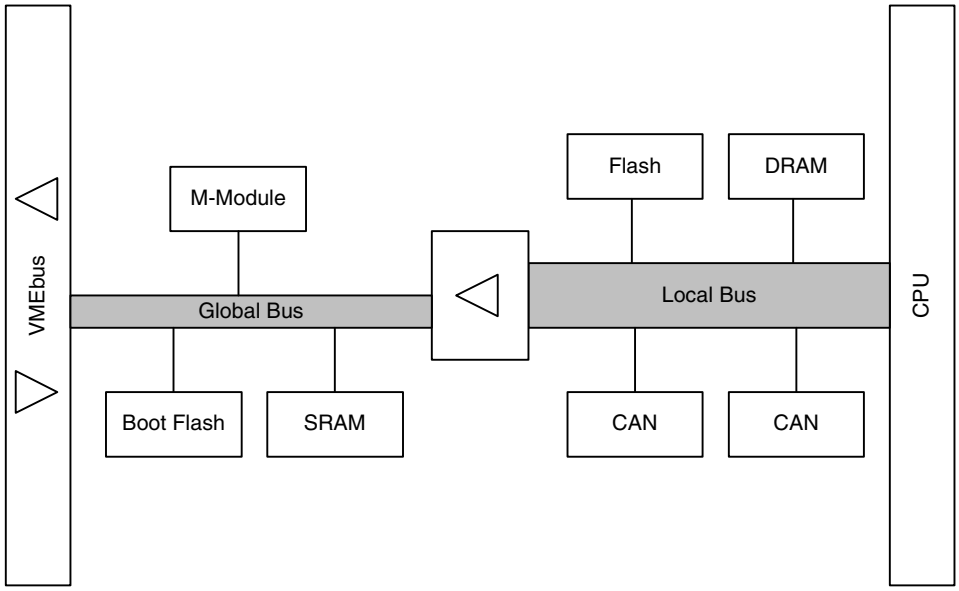
Switching from mains to battery power is performed automatically with no loss of data.

Battery voltage is applied to the +5VSTDBY line of the VMEbus connector.

2.2 Bus Structure

The B5 has two internal busses: a "local" 32-bit high-speed data bus and a "global" 16-bit bus.

Figure 5. *Bus Structure*



The local bus is not arbitrated, while the global bus is arbitrated according to the round-robin principle. All units connected to the global bus are quasi-dual-ported, i.e. they can be accessed both from the VMEbus and from the local bus. When the CPU and the VMEbus want to access one of the units simultaneously, access is granted alternatively. Read-modify-write accesses cannot be halted.

2.3 Processor Core

The B5 supports the principle of scalable CPU performance. Depending on the application, the user can choose between 4.5 MIPS and more than 100 MIPS of computing performance.

Generally, the following configurations are possible:

- MC68360 with CPU32+
- MC68360 plus MC68040V (CPU32+ disabled)
- MC68360 plus MC68060 (CPU32+ disabled)

This scalability is made possible by a special feature of the MC68360 microcontroller used on the board. In addition to a myriad of extremely powerful peripheral functions, this microcontroller has an on-chip "CPU32+". This central processing unit is completely user-mode compatible with the 68000 family. Its computing performance is 4.5 MIPS, which roughly equals the well-known MC68EC030. In order to reduce the chip surface, the manufacturer dispensed with caches, an FPU and an MMU. However, the performance of the CPU32+ is absolutely adequate for many cost-critical applications.

This CPU32+ on the MC68360 can be disabled at reset through the state of a number of signals. It can now be replaced by an external MC68040V or MC68060. In terms of software, the user hardly faces any changes - all peripheral functions of the MC68360 are available without any restrictions, but the computing performance has been multiplied several times: approx. 20 MIPS with the M68040V and up to more than 100 MIPS with the MC68060 (see manufacturers' data sheets). Depending on the CPU type, there are data and instruction caches and an FPU (except V type) and an MMU now. While retaining user-code compatibility, this concept opens up countless applications for the B5. The board can virtually grow with its application.

Table 3. *Compare Chart for MC68360, MC68040V and MC68060*

CPU/MHz	MIPS	FPU	MMU	Cache
68360/33	4.5	No	No	None
68040V/33	20	No	Yes	4k instr., 4k data
68060/50	100	Yes	Yes	8k instr., 8k data

Note: MEN gives no warranty on functionality and reliability of the B5 if you use any other processor than that supplied by MEN. Please contact either MEN directly or your local MEN sales office!

2.4 Memory

Memory of the B5 is divided into four parts:

- Boot Flash (global bus)
- Static, battery-backed RAM (global bus)
- Flash (for application software, local bus)
- Dynamic RAM (local bus)

In addition there is a serial EEPROM containing factory data and a user-programmable area.

2.4.1 Boot Flash (Global Bus)

The B5 has 512KB boot Flash with a 16-bit data bus.

Boot Flash contains the bootstrap loader based on MENMON. The Flash components used are divided into sectors. Two sectors (2 x 64KB) are used as boot sectors, where the most important hardware initializations such as VIC setup or configuration of the CPU clock frequency are made. All other sectors are available for the user.

2.4.1.1 Boot Flash Update from the Local CPU

The CPU can make memory-mapped accesses to the entire Flash, except for the boot sector. MENMON offers a function to update Flash (see Chapter 3.5 Updating Flash/Serial EEPROM on page 46).

2.4.1.2 Boot Flash Update from the VMEbus

Boot Flash can also be updated from the VMEbus, but only in the A24 address range. You can address the entire Flash without any additional hardware. The boot sector is write-protected. (See also Chapter 3.5 Updating Flash/Serial EEPROM on page 46.)

To keep the A24 slave address range occupied by the B5 small, the board uses the 1MB address range of SRAM for programming. During this time no access to SRAM is possible from the VMEbus. Mapping is changed through a bit in Global Control Register 1 and can also be dynamical (see Chapter 4.3.2 Global Control Registers on page 59).

The local CPU can be set to reset state during programming. This is defined by a bit in Global Control Register 0 (see Chapter 4.3.2 Global Control Registers on page 59).

2.4.2 Static RAM (Global Bus)

The B5 has two JEDEC SMD sockets for 1MB of static RAM. The RAM is organized in 16-bit steps.

You cannot access the SRAM when programming the boot Flash, since the SRAM address area at the VMEbus will be mapped to boot Flash (see Chapter 2.4.1 Boot Flash (Global Bus) on page 22).

The static RAM is battery-backed (see Chapter 2.1.2 SRAM and Real-Time Clock Battery on page 19).

2.4.3 Flash (Local Bus)

There are two blocks of Flash memory on the B5 with a size of 4MB each and a data bus width of 32 bits.

As standard, 4MB are mounted. Burst accesses of the MC68040/60 are possible and data can be cached.

Flash can be used to directly run user applications or to store compressed user applications, which are decompressed before the operating system is started and are transferred to DRAM. It is not possible to program and execute a user application at the same time.

You can update Flash on the B5 through a MENMON utility via a serial interface from a PC or via the VMEbus (description see Chapter 3.5 Updating Flash/Serial EEPROM on page 46).

2.4.4 Dynamic RAM (Local Bus)

The max. 36MB of DRAM are used by the B5 as local RAM memory.

4MB are permanently soldered. The MC68360 performs accesses to this DRAM without any wait states (3 clock cycles) for the CPU32+. Fast-page mode is supported for the CPU32+. With a "page hit" access time is reduced to 2 clock cycles. Fast-page mode is not supported for the MC68040/60, but burst mode is. During the power-up phase, MENMON sets a 2-1-1-1 burst. Each mode guarantees optimum performance for the respective CPU type.

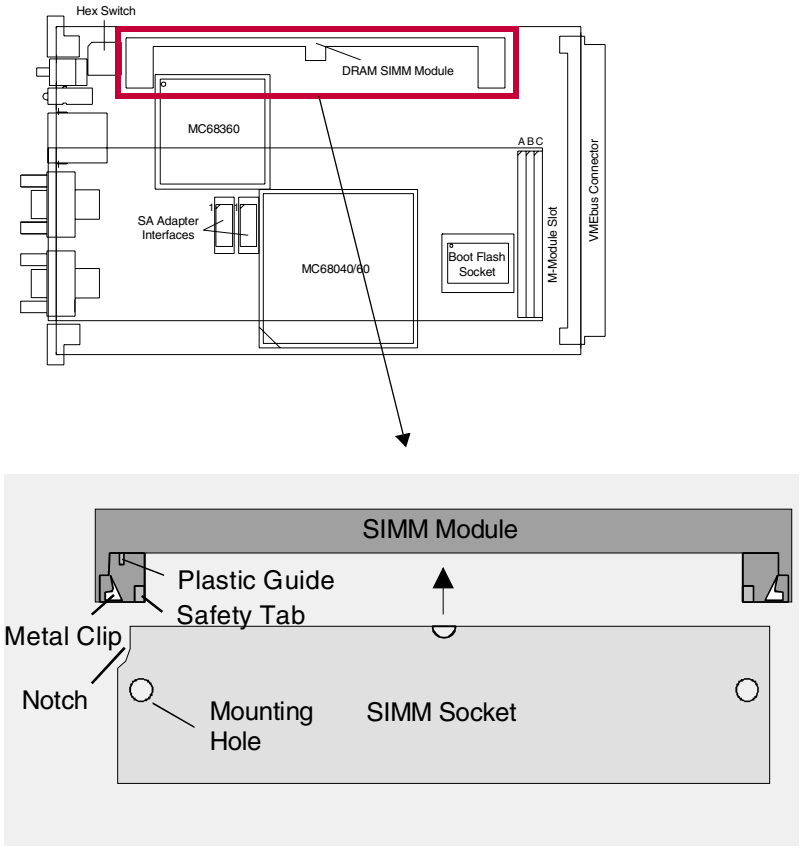
For DRAM extension, the B5 has a SIMM socket (location and installation see below). MENMON recognizes the size and access time of the DRAM SIMM (see Chapter 3.4.1 DRAM/SRAM Recognition on page 45).

A reset of the B5 also affects the memory controller of the MC68360. This means that there is no refresh of the DRAM as long as there is a reset condition or MENMON has not initialized the memory controller. Thus, data retention after a reset cannot be guaranteed.

2.4.4.1 PS/2 SIMM Installation

The B5 is normally shipped without any DRAM SIMM module installed. To install a PS/2 SIMM module, please stick to the following procedure.

Figure 6. *DRAM Installation*



The DRAM SIMM module will only fit as shown above because of a safety tab on one end of the SIMM socket which requires a notch in the SIMM module.

- ☑ Place the memory module into the socket at a 45° angle and make sure that all the contacts are aligned with the socket.
- ☑ Carefully press the memory module down until it clicks into place.
- ☑ The plastic guides must go through the two mounting holes on the sides and the metal clips must snap.
- ☑ To release the memory module, squeeze both metal clips outwards and carefully pull the module out of the socket.

2.4.4.2 Supported PS/2 SIMM Modules

You can install SIMM modules that support fast-page mode and 2K refresh, and write to single bytes (four CAS lines), e.g. MSC 9324200T3SS-6 (16MB).

Note: MEN gives no warranty on functionality and reliability of the B5 if you use any other module than that qualified and/or supplied by MEN. Please contact either MEN directly or your local MEN sales office.

2.4.5 Serial EEPROM

The B5 has a 4-kbit serial EEPROM, 1kbit of which is used for factory data. You can modify several MENMON settings through corresponding MENMON commands (see Chapter 3.6.3.2 Serial EEPROM Commands (EE-xxx) on page 49).

The remaining 3kbit are user-programmable. MEN provides a MENMON utility to write user data into the serial EEPROM. Chapter 3.5 Updating Flash/Serial EEPROM on page 46 gives a detailed description.

2.5 Ethernet Interface

SCC1 of the MC68360 is used as an Ethernet interface, supporting a data rate of 10Mbits/s. A DMA channel supports data transfer both when receiving and transmitting. Only half-duplex operation is possible.

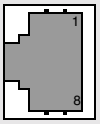
2.5.1 Connection

A standard 10Base-T RJ45 connector is available at the front panel.

Connector types:

- Modular 8/8-pin mounting jack according to FCC68
- Mating connector:
  Modular 8/8-pin plug according to FCC68

Table 4. *Pin Assignment of the 8-pin RJ45 Ethernet 10Base-T Connector*



1	TPO+
2	TPO-
3	TPI+
4	-
5	-
6	TPI-
7	-
8	-

Table 5. *Signal Mnemonics of the Ethernet 10Base-T Connector*

Name	Direction	Function
TPO+/-	out	Differential pair of transmit data lines
TPI+/-	in	Differential pair of receive data lines

2.5.2 General

Ethernet is a local-area network (LAN) protocol that uses a bus or star topology and supports data transfer rates of 100Mbps and more. The Ethernet specification served as the basis for the IEEE 802.3 standard, which specifies the physical and lower software layers. Ethernet uses the CSMA/CD access method to handle simultaneous demands. It is one of the most widely implemented LAN standards.

Ethernet networks provide high-speed data exchange in areas that require economical connection to a local communication medium carrying bursty traffic at high-peak data rates.

A classic Ethernet system consists of a backbone cable and connecting hardware (e.g. transceivers), which links the controllers of the individual stations via transceiver (transmitter-receiver) cables to this backbone cable and thus permits communication between the stations.

2.5.3 10Base-T

10Base-T is one of several adaptations of the Ethernet (IEEE 802.3) standard for Local Area Networks (LANs). The 10Base-T standard (also called Twisted Pair Ethernet) uses a twisted-pair cable with maximum lengths of 100 meters. The cable is thinner and more flexible than the coaxial cable used for the 10Base-2 or 10Base-5 standards. Since it is also cheaper, it is the preferable solution for cost-sensitive applications.

Cables in the 10Base-T system connect with RJ45 connectors. A star topology is common with 12 or more computers connected directly to a hub or concentrator.

The 10Base-T system operates at 10Mbps and uses baseband transmission methods.

2.6 CAN Bus Interfaces (Local Bus)

The B5 has two CAN bus interfaces controlled by two i82527 CAN controllers, supporting Standard CAN, Extended CAN, Full CAN and Basic CAN.

The physical interface is ISO 11898 High Speed.

2.6.1 General

CAN bus provides an open fieldbus system for industrial applications.

Its primary characteristics are:

- Bus length up to 1,000m
- Transfer rates: 62.5kbits/s..1Mbits/s
- High immunity to external and internal errors
- Short message lengths (0..28 bytes)
- Short transfer delays due to short messages

CAN allows multimaster access according to the CSMA/CA principle (Carrier Sense Multiple Access with Collision Avoidance) with bitwise arbitration depending on the message priority. If two or more network participants want to access the bus simultaneously, it will always be the most important message that is transmitted first. This avoids loss of transmission time.

The transfer rate depends on the line length:

Table 6. *CAN Bus Transfer Rates related to Line Lengths and Cables*

Transfer Rate	Line Length	Recommended Cables[1]
1Mbits/s	40m	0..40m: 0.25mm^2, 0.34mm^2 AWG23, AWG22
500kbits/s	100m	40..300m: 0.34mm^2..0.6mm^2 AWG22, AWG20
125kbits/s	500m	300..600m: 0.5mm^2, 0.6mm^2 AWG20
62.5kbits/s	1,000m	600..1,000m: 0.75mm^2, 0.8mm^2 AWG18

[1] Length and cross section

At bit rates lower than 1Mbits/s the bus length may be lengthened significantly. A data rate of 62.5kbits/s allows a bus length of 1,000m. ISO 11898 compliant transceivers specify max. bus length of about 1,000m. However, it is allowed to use bridge devices or repeaters to increase the allowed distance between ISO 11898 compliant nodes to more than 1,000m.

2.6.2 Basic CAN, Full CAN and Extended CAN

CAN exists in two forms; a basic CAN and a higher form with an "acceptance filter". **Basic CAN** has a tight coupling between the CPU and the CAN controller, where all messages broadcast on the network have to be individually checked by the microcontroller. This results in the CPU being "tied up" checking messages rather than processing them, all of which tends to limit the practicable baud rate to 250kbaud. The introduction of an acceptance filter masks out the irrelevant messages, using identifiers (ID) and presents the CPU with only those messages that are of interest. This is usually referred to as **Full CAN**. The Full CAN protocol allows for two lengths of identifiers: part A allows for 11 message identification bits, which yield 2,032 different identifiers (16 are reserved), while **Extended CAN** (part B) has 29 identification bits, producing 536,870,912 separate identifiers.

2.6.3 Implementation on the Board

Access to the CAN controllers is via the local 32-bit bus. The CAN controllers support only a 16-bit data bus, so that only byte or word accesses can be made to the controllers' registers.

Figure 7. *Byte Ordering for CAN Controllers*

32..24	23..16	15..8	7..0
Not used	Not used	CAN	CAN

The MC68360 supports an interrupt for each of the CAN controllers.

The two interfaces are optically isolated from each other and from the system.

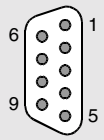
2.6.4 Connection

The CAN bus is connected by means of two 9-pin micro D-Sub connectors at the front panel.

Connector types:

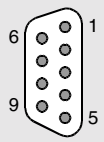
- Pin connector with locking post, ITT Cannon MDSM-9PE-Z10-VR
- Mating connector:
  9-pin micro D-Sub socket connector with screw locking, ITT Cannon MDSM-9SC-Z11-VS1

Table 7. *Pin Assignment of the 9-pin micro D-Sub CAN A Plug Connector*



6	GND_A	1	-
7	CAN_A_H	2	CAN_A_L
8	-	3	GND_A
9	-	4	CAN_L_RES
		5	CAN_H_RES

Table 8. *Pin Assignment of the 9-pin micro D-Sub CAN B Plug Connector*



6	GND_B	1	-
7	CAN_B_H	2	CAN_B_L
8	-	3	GND_B
9	-	4	CAN_L_RES
		5	CAN_H_RES

Table 9. *Signal Mnemonics of the CAN Bus Connectors*

Name	Direction	Function
CAN_H_RES	-	CAN_H for connection with CAN_L_RES[1]
CAN_L_RES	-	CAN_L with termination resistor[1]
CAN_*x*_H	in/out	Non-inverted data
CAN_*x*_L	in/out	Inverted data
GND_*x*	-	Ground

[1] You can provide the CAN bus with the necessary termination resistors by connecting pins *CAN_L_RES* and *CAN_H_RES*.

2.7 Serial Interfaces

2.7.1 RS232 Interfaces

The CPM of the MC68360 has two Serial Management Controllers (SMCs). Both operate in UART mode. On the B5 they serve two serial RS232 interfaces. They support UART protocols and have two DMA channels each to transmit data.

UART6 does not use any handshake lines, UART5 supports RTS and CTS.

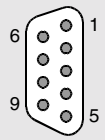
The interfaces support up to 115.2kbit/s. They are optically isolated from each other and from the system and are led to a 9-pin micro D-Sub connector at the front panel.

Because of optical isolation, the pin assignment is not completely PC-compatible. Only lines *RxD5*, *TxD5* and *GND5* comply with PC assignment.

Connector types:

- Pin connector with locking post, ITT Cannon MDSM-9PE-Z10-VR
- Mating connector:
  9-pin micro D-Sub socket connector with screw locking, ITT Cannon MDSM-9SC-Z11-VS1

Table 10. *Pin Assignment of the 9-pin micro D-Sub RS232 Plug Connector*



6	RxD6	1	CTS5
7	TxD6	2	RxD5
8	-	3	TxD5
9	GND6	4	RTS5
		5	GND5

Table 11. *Signal Mnemonics of the RS232 Interfaces*

Name	Direction	Function
CTS*x*	in	Clear to Send
GND*x*	-	Ground
RTS*x*	out	Request to Send
RxD*x*	in	Receive data
TxD*x*	out	Transmit data

Note: MEN supplies an adapter cable for the 9-pin micro D-Sub connectors (see MEN's website). This cable leads the 9-pin micro D-Sub receptacle connector to a standard 9-pin D-Sub plug connector. Pin assignment stays the same. You may need, however, an additional "null modem" cable for connection of a terminal or PC.

2.7.2 ISO 9141-2/RS232 Interface

The serial ISO 9141-2 interface is served by SCC3 of the MC68360. This channel supports synchronous protocols in addition to UART protocols and has one DMA channel for transmission of data.

The interface uses the *RTS* line of the SCC for implementation of the *L-line* of the ISO 9141-2 interface. Lines *RxD* and *TxD* form the *K-line*.

The ISO 9141-2 interface supports baud rates of 5bit/s..115.2kbit/s. Timer GPT4 of the MC68360 is used for baud rate generation (see Chapter 4.4 Implementation of MC68360 on page 61).

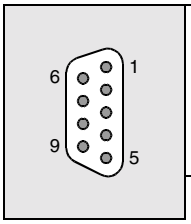
The interface is optically isolated from the system and led to a 9-pin micro D-Sub connector at the front panel.

Alternatively, you can use the port as an RS232 interface without handshake lines. To use the interface as RS232, do not connect pins 1 (*VD*), 7 (*L-line*) and 8 (*K-line*). This interface will support up to 115.2kbit/s.

Connector types:

- Pin connector with locking post, ITT Cannon MDSM-9PE-Z10-VR
- Mating connector:
  9-pin micro D-Sub socket connector with screw locking, ITT Cannon MDSM-9SC-Z11-VS1

Table 12. *Pin Assignment of the 9-pin micro D-Sub ISO 9141-2 Plug Connector*



6	-	1	VD
7	L-line	2	RxD3
8	K-line	3	TxD3
9	-	4	-
		5	GND

Table 13. *Signal Mnemonics of the ISO 9141-2 Interface*

Name	Direction	Function
GND	-	Ground
K-line	in/out	Bidirectional data
L-line	out	Output
RxD3	in	Alternative UART 3 receive data
TxD3	out	Alternative UART 3 transmit data
VD	in	External supply voltage

2.7.3 Optional Serial Interfaces

Two further, user-configurable ports are available on the B5. Both, however, have no physical interface. This is implemented using interface adapters via a ribbon cable, depending on the user's requirements. A large variety of standard interface adapters is available with or without optical isolation (e.g. RS232, RS485, TTY etc., see MEN's website).

For a detailed functional description of MEN's SA adapters, please refer to the corresponding hardware manual.

Both interfaces are implemented using an SCC. UART mode is supported; other operating modes, e.g. HDLC etc., can be implemented. Four handshake lines are available. Physical line assignment is done using interface adapters. SCC2 is led to P9, SCC4 is led to P10.

Note: P9 can also be used as an Ethernet interface. Signal assignment allows operation of the MC68360 with two Ethernet interfaces. However, you need a special interface adapter. See MEN's website for adapters.

Connector types:

- Pin contact strips similar to IDC ribbon-cable connector according to DIN41651/ MIL-C-83503 but without housing and lock
- Mating connector:
  10-pin receptacle, available with or without tension relief for ribbon-cable connection, 1.27mm pitch

Figure 8. *Position of P9 and P10*

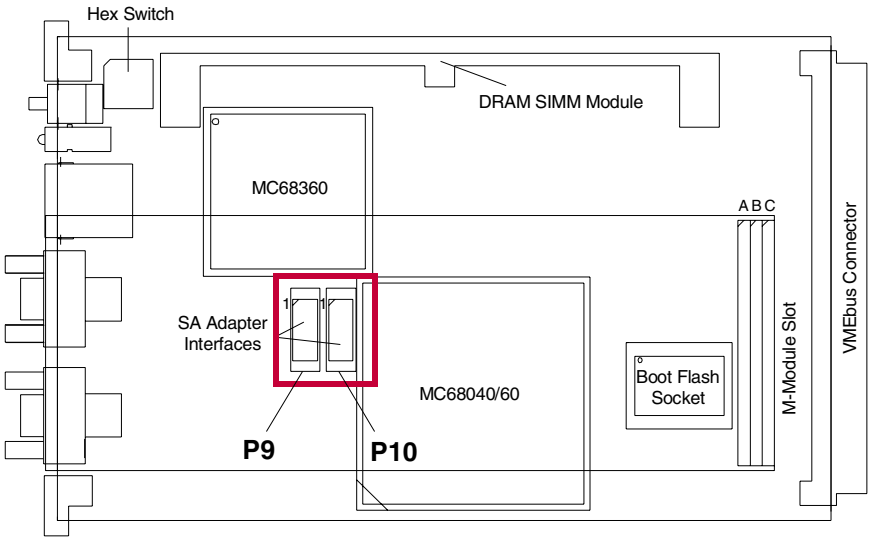
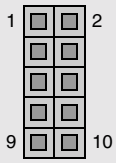
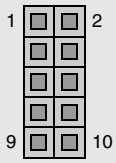


Table 14. *Pin Assignment of the 10-Pin Plug Connector P9*



1	GND	2	VCC
3	TxD2	4	RxD2
5	DTR2	6	RTS2/TEN2
7	DSR2/TCLK2	8	CTS2/COL2
9	DCD2/RENA2	10	RI2/RCLK2

Table 15. *Pin Assignment of the 10-Pin Plug Connector P10*



1	GND	2	VCC
3	TxD4	4	RxD4
5	DTR4	6	RTS4
7	DSR4	8	CTS4
9	DCD4	10	-

Table 16. *Signal Mnemonics of the Optional Serial Interfaces*

Name	Direction	Function
CTS*x*	in	Clear to Send
DCD*x*	in	Data Carrier Detect
DSR*x*	in	Data Set Ready
DTR*x*	out	Data Terminal Ready
GND	-	Ground
RI*x*	in	Ring Indicator
RTS*x*	out	Request to Send
RxD*x*	in	Receive data
TxD*x*	out	Transmit data
VCC	out	5V supply voltage
COL2	in	Alternative Ethernet: collision
RCLK2	in	Alternative Ethernet: receiver clock
RENA2	in	Alternative Ethernet: receiver enable
TCLK2	in	Alternative Ethernet: transmitter clock
TEN2	out	Alternative Ethernet: transmit enable

2.7.3.1 Installation of SA Adapters

Please refer to Chapter 1.4 Installing an M-Module and SA Adapter on page 16 on how to install one SA adapter. If you want to install a second SA adapter, contact MEN's technical support at support@men.de.

2.8 M-Module Slot (Global Bus)

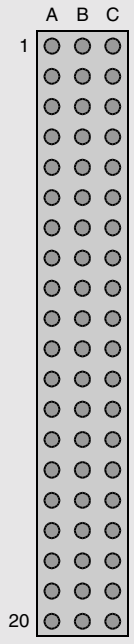
The M-Module slot enables the user to add a number of input/output functions to the B5 CPU board. The wide range of standardized M-Modules includes not only process I/O modules but also interface extensions, network boards (such as Profibus etc.), DSP and transputer modules and special-purpose functions.

The B5 has one M-Module slot. Peripherals can be connected only to the front-panel connector of the M-Module. The 24-pin peripheral receptacle connector of the M-Module is not supported.

2.8.1 Connection

The signals from the B5 are fed to the M-Module via three 20-pin receptacle connector rows. These connectors correspond to connectors on the M-Module. The pin assignment corresponds to the M-Module specification (see Chapter 5.1 Literature and WWW Resources on page 64).

Table 17. *Pin Assignment of the 60-Pin M-Module Receptacle Connector*



	A	B	C
1	/CS	GND	/AS
2	A01	+5V	D16
3	A02	+12V	D17
4	A03	-12V	D18
5	A04	GND	D19
6	A05	/DREQ	D20
7	A06	/DACK	D21
8	A07	GND	D22
9	D08/A16	D00/A08	TRIGA
10	D09/A17	D01/A09	TRIGB
11	D10/A18	D02/A10	D23
12	D11/A19	D03/A11	D24
13	D12/A20	D04/A12	D25
14	D13/A21	D05/A13	D26
15	D14/A22	D06/A14	D27
16	D15/A23	D07/A15	D28
17	/DS1	/DS0	D29
18	/DTACK	/WRITE	D30
19	/IACK	/IRQ	D31
20	/RESET	SYSCLK	/DS2

2.8.2 Addressing the M-Module

The M-Module is connected to the global CPU bus, i.e. it can be accessed both from the local CPU and from the VMEbus.

The B5 supports the following M-Module characteristics:

- A08, A24, D16, INTA, INTC
- D32, TRIGA, TRIGB, DMA (only for local CPU)

The M-Module occupies a maximum 1MB address space with A24 access (see Chapter 4.1 Address Mappings on page 55).

For other VMEbus masters only data access and interrupts are possible. TRIGA, TRIGB and DMA can be used only by the local CPU.

2.8.3 DMA

The DMA lines are handled by the MC68360. The M-Module timing in DMA mode does **not** comply with the M-Module Specification.

Note: If you want to use DMA mode, please contact MEN's technical support at support@men.de.

2.8.4 Interrupts

The M-Module may generate interrupts **either**

- to the local CPU **or**
- to an external interrupt handler via the VMEbus.

An interrupt cannot be handled by the local CPU and the VMEbus at once. The interrupt handler must be defined through Global Control Register 2 (M-Module IRQ Mode, read/write, offset 0x05).

The interrupt must be enabled in Global Control Register 3 (M-Module IRQ Mask, read/write, offset 0x07).

Interrupts to the Local CPU

INTA and INTC interrupts are possible in accordance with the M-Module Specification. With INTA interrupts to the local CPU the MC68360 generates an autovector 2. With INTC interrupts the M-Module supplies the interrupt vector in the interrupt acknowledge cycle.

Interrupts via the VMEbus

The VIC generates M-Module interrupts via the VMEbus to an external interrupt handler. To trigger an interrupt request at the VMEbus, you must enable the corresponding VMEbus IRQ in Global Control Register 2 (M-Module IRQ Mode, read/write, offset 0x05).

INTA and INTC interrupts are possible in accordance with the M-Module Specification. With INTA interrupts the interrupt vector is generated by the VIC during the IACK cycle at the VMEbus. With INTC interrupts the M-Module supplies the interrupt vector when a read access is made to the IACK address range (see Chapter 4.1 Address Mappings on page 55).

When the interrupt was initiated at the VMEbus, it is disabled and must be re-enabled in the interrupt service routine by the user through Global Control Register 3 (M-Module IRQ Mask, read/write, offset 0x07).

2.9 VMEbus Interface

The B5's VMEbus interface conforms to the ANSI/IEEE STD 1014-1987, IEC 821 and 297 specifications.

The VMEbus interface of the B5 has the following features:

- VIC068A VMEbus chip
- CY7C964 bus drivers
- Slot-1 functionality
- Master interface:
  - Requester
  - A24-, A16-, D16-transfer modes
  - Block transfers
  - Read-modify-write (not supported with MC68060)
- Slave interface:
  - A24, A16, D16 transfers
  - Block transfers
- Interrupt handler: 4-level
- Interrupter: 7-level
- Interprocessor communication facilities

2.9.1 Slot-1 Function

The slot-1 function will be autodetected at power-up. B5 has the following functions as a VMEbus system controller:

- *SYSRESET#* generation
- *SYSCLK* generation
- Bus arbitration
- Bus arbitration timeout
- Bus transfer timeout

The bus arbitration and transfer timeouts can be modified using MENMON, see Chapter 3.6.3.3 Commands for VMEbus Interface Parameters on page 50.

2.9.2 VMEbus Master Interface

The B5 supports the A16 and A24 address ranges of the VMEbus. D08(E/O), D16 and D16BLK data transfers as well as read-modify-write (RMW) accesses can be performed[1]. Block transfers use only global SRAM.

[1] RMW access is not supported with MC68060.

2.9.3 VMEbus Slave Interface

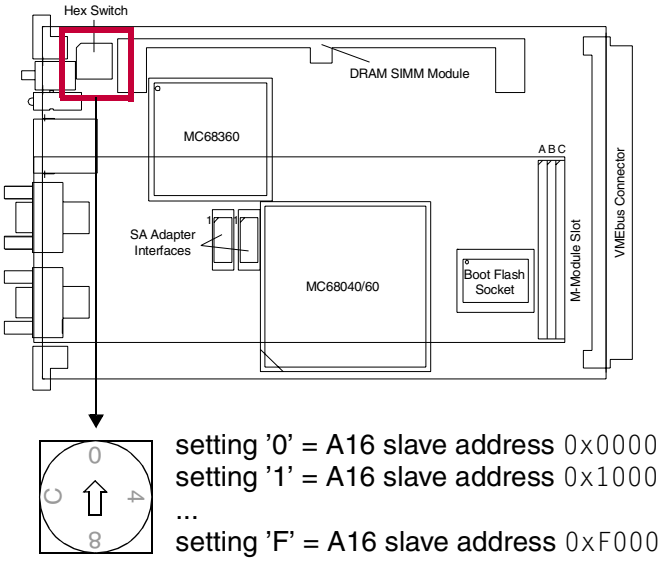
A VMEbus master can access the global bus of the B5. Accesses are possible in the A16 or A24 address range of the VMEbus.

Address maps for A16 and A24 access are included in the appendix (Chapter 4.1.2 Global Bus Access by Another VMEbus Master on page 56).

2.9.3.1 Setting the Slave Address

The onboard hex switch determines the A16 slave address for the B5. MENMON reads the hex switch and writes the respective value to Local Control Registers 0 and 1 (see Chapter 4.3.1 Local Control Registers on page 58).

Figure 9. *Hex Switch for VMEbus A16 Slave Address*



You can set the A24 slave address in MENMON (see Chapter 3.6.3.3 Commands for VMEbus Interface Parameters on page 50).

Note: Because of their 32-bit structure Local Control Registers 0 and 1 cannot and must not be written to by another VMEbus master.

2.9.4 VMEbus Interrupt Handler

The B5 supports VMEbus interrupt levels 3..6. The VIC chip maps incoming VMEbus interrupts to the local interrupt lines. On-board logic then generates an interrupt request for the CPU at the free interrupt request lines 3 and 5.

This would allow only two interrupt levels from the VMEbus to the CPU. However, on-board logic encodes interrupts 4 and 6 at levels 3 and 5. To permit the VMEbus to access the right interrupter in its *IACK#* cycle, the levels are decoded again.

Table 18. *VMEbus Interrupt Levels*[1]

CPU Level	Description
5	VMEbus level 6, VMEbus level 5
3	VMEbus level 4, VMEbus level 3

2.9.5 Utility Bus Signals *ACFAIL#* and *SYSFAIL#*

Lines *ACFAIL#* and *SYSFAIL#* of the VMEbus can generate an interrupt to the MC68360 at level 7 through the VIC controller.

2.9.6 VMEbus Interrupter

The local CPU can generate interrupts on the VMEbus at any level between 1 and 7. The Interrupt Request Register and the related Interrupt Request Vector Base Register of the VIC must be programmed accordingly.

With the help of on-board logic, even an M-Module can generate any interrupt on the VMEbus, without support by the local CPU (see Chapter 2.8.4 Interrupts on page 37).

2.9.7 *Remote2* Line

The *Remote2* line is a VMEbus output (*SERCLK*), which is controlled through two ports of the MC68360: one port controls the level, another activates or deactivates the driver.[2] *SERCLK* is a buffered signal that complies with the VMEbus specification.

2.9.8 Stand-Alone Operation

The B5 is prepared for stand-alone operation. For this reason, some VMEbus lines were provided with 10kΩ pull-up resistors. This does not comply with the VMEbus specification!

Pulled-up lines:
ACFAIL#, BR3#, BR2#, BR1#, BR0#, IRQ7#, IRQ6#, IRQ5#, IRQ4#, IRQ3#, IRQ2#, SYSFAIL#, BERR#, SYSRESET#, AS#, IACK#.

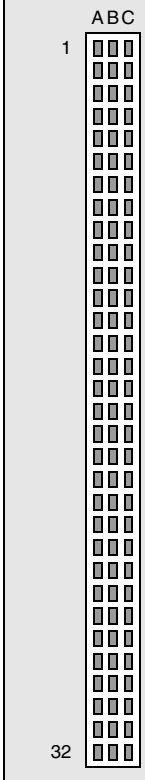
[1] See also Chapter 4.2 Interrupt Handling on page 57.
[2] See also Chapter 4.4 Implementation of MC68360 on page 61.

2.9.9 Connection

Connector types:

- Type-C plug connector according to DIN41612/MIL-C-55302/IEC603-2
- Mating connector:
  Type-C 96-pin receptacle according to DIN41612/MIL-C-55302/IEC603-2, available with solder/wire-wrap pins, for hand-soldering connection or for insulation piercing connection (IDC)

Table 19. *Pin Assignment of the 3-Row, 96-Pin VMEbus Connector P1*



	A	B	C
1	D0	BBSY#	D8
2	D1	BCLR#	D9
3	D2	ACFAIL#	D10
4	D3	BG0IN#	D11
5	D4	BG0OUT#	D12
6	D5	BG1IN#	D13
7	D6	BG1OUT#	D14
8	D7	BG2IN#	D15
9	GND	BG2OUT#	GND
10	SYSCLK	BG3IN#	SYSFAIL#
11	GND	BG3OUT#	BERR#
12	DS1#	BR0#	SYSRST#
13	DS0#	BR1#	LWORD#
14	WRITE#	BR2#	AM5
15	GND	BR3#	A23
16	DTACK#	AM0	A22
17	GND	AM1	A21
18	AS#	AM2	A20
19	GND	AM3	A19
20	IACK#	GND	A18
21	IACKIN#	Remote2[1]	A17
22	IACKOUT#	-	A16
23	AM4	GND	A15
24	A7	IRQ7#	A14
25	A6	IRQ6#	A13
26	A5	IRQ5#	A12
27	A4	IRQ4#	A11
28	A3	IRQ3#	A10
29	A2	IRQ2#	A9
30	A1	IRQ1#	A8
31	-12V	+5VSTDBY	+12V
32	+5V	+5V	+5V

[1] SERCLK is used as Remote2 on the B5. Please refer to Chapter 2.9.7 Remote2 Line on page 40.

2.10 Real-Time Clock

The real-time clock of the B5 features a built-in quartz crystal, time and date function, and CMOS circuitry for low power consumption.

Battery buffering is identical to that of the SRAM (cf. Chapter 2.1.2 SRAM and Real-Time Clock Battery on page 19). The RTC is located at the global bus but is not accessible from the VMEbus.

2.11 User LEDs

The four front LEDs of the B5 are entirely user-programmable through Global Control Register 4 (LED, read/write, offset 0x0D).

2.12 Reset/Abort Button

The reset button at the front panel triggers a reset. If the slot-1 function is active, this reset will act globally for the VMEbus (*SYSRESET#*). If the slot-1 function is not active, the reset will act locally. After the reset the CPU will detect the cause of the reset, i.e. power-on, external reset, watchdog or software reset by reading status registers.

The abort button activates the non-maskable interrupt of the CPU via the on-board logic at level 7. The abort interrupt can be masked through Local Control Register 4 (Interrupt 7 Mask Register, read/write, 0x00FE0009).

2.13 Watchdog Logic

The B5 has an external hardware watchdog. It can operate in one of two modes:

- Mode 1: normal watchdog triggered by CPU signal
- Mode 2: watchdog triggered by software write access to Local Control Register 2 (Watchdog Mode/Status Register, read/write, 0x00F80008)

If the watchdog is not triggered every 70ms, it will cause a reset. At the latest, the reset will be caused 140ms after the last trigger signal. After a reset the user software can detect the state of the watchdog by reading Local Control Register 2 (Watchdog Mode/Status Register, read/write, 0x00F80008).

2.14 On-Board Bus Error Logic

An on-board bus error logic on B5 monitors all accesses of the CPU to the local or global bus. If an access has not been completed after 15µs, the logic generates a bus error.

Accesses of B5 to the VMEbus and slave accesses to B5 by other VMEbus masters are terminated externally through the VMEbus slot-1 function (see Chapter 2.9.1 Slot-1 Function on page 38).

3 The MENMON Debugger

3.1 General

MENMON is a simple assembly-language debugger for members of Motorola's 68k family. It provides a simple user console interface and can easily be extended and ported.

Purpose

- Debugging applications without any operating system
- Bootstrapping operating systems
- Hardware testing

Features

- Needs only small amount of memory
- Single stepping, breakpoints
- Change/examine memory
- Line-by-line assembler/disassembler
- Download Motorola S-records
- Memory testing
- Exception reporting

Supported Processors

MENMON supports the following members of Motorola's 68k family:

- 68000 core (e.g. 68000, 68008, 68302)
- 68020 core (e.g. 68020, 68030)
- CPU32/CPU32+ core (e.g. 68331, 68332, 68360)
- 68040, 68060

3.2 Console

MENMON uses the console to communicate with the user. On most systems, a serial line is used. However, some implementations may provide the console interface as graphics video display, for example.

If a serial line is used, the baud rate used depends on the implementation. In systems with nonvolatile memory the baud rate can usually be changed using a special MENMON command. The default configuration should be 9600 baud/8n1 on all systems.

Systems without nonvolatile memory may detect the baud rate automatically. In this case, the user must enter a <CR> (ASCII 0xD) so that MENMON can guess the baud rate from the character received.

MENMON uses the serial debug interface (SMC1).

3.2.1 Serial Interface

The serial interface can be set to the following values using commands *EE-BAUD* and *EE-PAR*:

Baud rate: 4800, 9600, 19200, 38400 baud
Parity: 7e1, 7e2, 8n1

3.3 B5 MENMON Memory Map

MENMON uses the following memory areas:

Table 20. *MENMON Memory Map*

Address Range	Memory	Function
0x 0000 0000..0000 FFFF	ROM	Primary MENMON code
0x 0001 0000..0001 FFFF	ROM	Secondary MENMON code
0x 00FF E100..00FF E111	68360 RAM	Buffer descriptors for serial console
0x 00FF E220..00FF E3FF	68360 RAM	Stack
0x 00FF E400..00FF E56B	68360 RAM	Globals

The memory areas occupied by MENMON can be used for other purposes if MENMON is no longer forked up, otherwise these areas must remain intact.

3.4 MENMON Start-Up

Two MENMON versions reside in the boot Flash of the B5:

- Flash memory is programmed with primary MENMON at production. It is write-protected. Primary MENMON does nothing but call or re-program the secondary MENMON.
- Secondary MENMON is the current MENMON version and is forked up directly after the start of primary MENMON. Normally it immediately starts the OS bootstrap loader if EE-AUTOBOOT is set to '1'.

3.4.1 DRAM/SRAM Recognition

At start-up MENMON detects the DRAM and SRAM configuration. It recognizes the following DRAM SIMM configurations:

- 0, 4, 8, 16, 32 MB
- 60 or 70ns

The chip-select registers of the MC68360 are programmed with reference to the detected values. The setup of OR1, OR2 and OR6 reflects the DRAM configuration, OR3 the SRAM configuration.

3.5 Updating Flash/Serial EEPROM

MENMON provides the possibility of updating Flash devices and a user-programmable area of the serial EEPROM on the B5 via a serial interface or from a VMEbus master.

3.5.1 Update File

Updates are always done using an update file containing the target data.

The file extension determines the device and sector of Flash that the file is to be programmed to:

- <*filename*>.*Byy*
  File is copied to boot Flash starting at sector *yy*
- <*filename*>.*Fyy*
  File is copied to local Flash starting at sector *yy*
- <*filename*>.*Eyy*
  File is copied to serial EEPROM starting at sector *yy*

The sector number is a decimal value. You can calculate the start address from the decimal value according to the following table:

Table 21. *Flash Sector Sizes*

Flash Device	Sector Size
Boot Flash[1]	64KB
Local Flash (1 or 2 MB)	64KB
Local Flash (4 or 8 MB)	256KB
Serial EEPROM[2]	2 bytes

[1] You cannot program sector 0 of the boot Flash. It is write-protected.
[2] If you want to program the EEPROM and use the file extension to specify the start address, note that the highest start address you can state is `0xC6` (with extension *.E99*).

When a file is larger than one sector, the following sectors of the device will also be programmed.

If the start address is not the beginning of a block of Flash, the memory area from the beginning of the block to the start address will be erased.

The update file is transferred to DRAM before being programmed to Flash. The DRAM of the B5 must therefore be large enough for the entire download file. Download itself will require another 68KB, i.e. with 4MB of DRAM on the board, the update file may be max. 4032KB.

3.5.2 Update via a Serial Interface

You must connect a PC as a terminal to the B5 and start MENMON as described in Chapter 1.3 Integrating the Board into a System on page 15. You also need a terminal program on your PC (e.g. HyperTerminal under Windows 95).

Enter command *SERDL* (serial download) in MENMON. The terminal program must then transmit an update file using YModem protocol.

3.5.2.1 Transfer Times

Transfer of a 4MB file will take about 1:30h at 9,600 baud. There are two possibilities to shorten the transfer time:

- Change the baud rate using MENMON command *EE-BAUD 38400*, then reset the target system. MENMON as well as data transfer will then work at 38,400 baud.
- Change the baud rate temporarily using *SERDL 38400*. In this case the *SERDL* command must be entered at 9,600 baud. Then you must set the terminal program to 38,400 baud. Not all terminal programs will work with this method!

After download you must restart the B5.

3.5.3 Update via VMEbus

A VMEbus master can update Flash and serial EEPROM of the B5. To do this, the VMEbus master must run OS-9 and utility *b5_load*. The master needs to know the A16 and A24 VMEbus slave address of the B5.

When you have entered the *b5_load* command (see examplex below), *b5_load* triggers a reset of the B5 and writes a command to the B5 SRAM. At start-up, MENMON checks whether this command is in SRAM. If this is the case, MENMON and the *b5_load* utility will use a special protocol to transfer the file to the B5 block by block. To do this, they use the first 4KB of the B5 SRAM.

Unlike serial download, you may also use command line options to define to which device and address the data are to be programmed. (In this case the command will simply ignore the file extension.)

When the file was transferred completely the master CPU will trigger another reset of the B5, which will then start up again.

Examples

VMEbus Master: MEN A10 board
hex switch setting of B5: 7
VME-A24 slave address of B5: 0x000000

Programming file *file.F00* to local Flash of B5, offset 0:

```
$ b5_load -a=8c007000 -b=88000000 file.F00
```

Programming file *file* to local Flash of B5, offset 0x200000:

```
$ b5_load -a=8c007000 -b=88000000 -d=F -s=200000 file
```

3.6 MENMON User Interface

3.6.1 Command Line Editing

MENMON provides a rudimentary command line editor:

<CTRL> <H>	Backspace and delete previous character
<CTRL> <X>	Delete whole line
<CTRL> <A>	Retrieve last line

3.6.2 Numerical Arguments

Most MENMON commands require one or more arguments. Numerical arguments may be numbers or simple expressions:

<num>	*num* is interpreted as a hexadecimal value
$<num>	Same as above
#<num>	*num* is interpreted as a decimal value
%<num>	*num* is interpreted as a binary value
.<REG>	Use the value of register <REG>

These arguments can be combined using the arithmetic operators "+" and "-".

Example:[1]

```
MenMon> D 10000
```
Dumps address `0x10000`

[1] Some of the addresses used in our examples may not be suitable for your board's address mapping. If you want to try out MENMON's functions, please compare the example addresses with your mapping first!

3.6.3 Commands

3.6.3.1 General

BO — Call Bootstrap Loader

The *BO* command is used to start another operating system bootstrap loader. When no address is given, the address is taken from the serial EEPROM (this can be set using *EE-BS*).

Examples

`MenMon> BO`	Call default bootstrap loader. Take address from EEPROM.
`MenMon> BO 2000000`	Call bootstrap loader at address `0x2000000`

3.6.3.2 Serial EEPROM Commands (EE-xxx)

These commands are used to modify/display MENMON parameters in the serial EEPROM:

EE

Display all MENMON parameters in EEPROM

EE-DEF

Program MENMON parameters in EEPROM with defaults

EE-BAUD [<baud>]

Set/display console baud rate. Possible values: 4800, 9600, 19200, 38400 baud (default: 9600)

EE-PAR [<par>]

Set/display console parity. Possible values: 7e1, 7e2, 8n1 (default: 8n1)

EE-BS [<addr>]

Set/display bootstrap address

EE-PROD

Display production data

EE-DUMP

Hex dump of entire EEPROM

EE-AUTOBOOT [<0/1>]

Enable/disable automatic jump to OS bootstrap loader

SERDL

Flash update using YModem protocol via serial line (see Chapter 3.5 Updating Flash/Serial EEPROM on page 46).

3.6.3.3 Commands for VMEbus Interface Parameters

These parameters are stored in the serial EEPROM:

VME

Display VME controller parameters in EEPROM

VME-DEF

Program EEPROM with defaults

VME-IRQ [<mask>]

VMEbus IRQ levels enable mask (default: 0x78)

7	6	5	4	3	2	1	0
-	L6	L5	L4	L3	0	0	0

When an L*x* bit is set, MENMON enables the corresponding VMEbus interrupt level during startup. When the bit is cleared, MENMON sets up the corresponding VICR register but does not enable the interrupt. Bits marked '0' are ignored.

VME-TTR [<val>]

VIC Transfer Timeout Register (default: 0x48)
This setting corresponds to the VIC TTR register.

7	6	5	4	3	2	1	0
VMEbus TO			Localbus TO			0	IVA

These values define the timeout periods for the VMEbus and local bus timers (before a bus error is generated):
VMEbus TO and *Localbus TO* are defined as follows:
000 = 4µs
001 =16µs
010 = 32µs
011 = 64µs
100 = 128µs
101 = 256µs
110 = 512µs
111 = Infinite (timeout disabled)

IVA:
Include VMEbus acqusition time in local bus timer (see VIC068A manual for details).

VME-ICR [<val>]

VIC Interface Configuration Register (default: 0x44)
This setting corresponds to the VIC ICR register.

7	6	5	4	3	2	1	0
RM3..RM1			Deadlk		Met	Tur	0

Normally this register should not be modified!

VME-ARCR [<val>]

VIC Arbiter Configuration Register (default: 0x60)
This setting corresponds to the VIC ARCR register.

7	6	5	4	3	2	1	0
AM	ReqLev		0	Fairness Tmr			

AM defines the VMEbus arbitration mode:
0 = Round robin arbitration
1 = Priority arbitration

ReqLev defines the VMEbus request level:
00 = BR0
01 = BR1
10 = BR2
11 = BR3

Fairness Tmr:
0000 = Fairness disabled
0001..
1110 = Number times 2µs
1111 = Timeout disabled

VME-RCR [<val>]

VIC Release Control Register (default: 0x00)
This setting corresponds to the VIC RCR register.

7	6	5	4	3	2	1	0
RelMode		0	0	0	0	0	0

RelMode defines the VMEbus release mode:
00 = ROR: Release on Request
01 = RWD: Release when done
10 = ROC: Release on BLCR# assertion
11 = BCAP: VMEbus capture and hold

VME-A24SA [<val>]

VME A24 compare address (default: 0x00)
Defines the slave address of the B5 in A24 mode:

7	6	5	4	3	2	1	0
0	0	0	0	A23..A20			

These bits are compared with VMEbus lines A23..A20.
Do not enter odd numbers!
Slave access cannot be disabled!

Note: When the EEPROM contents are modified, the new parameters will have no effect until the system is restarted.

3.7 Exception Handling

MENMON catches all 68000 exceptions and interrupts. If an exception occurs while the user program is running (started using *GO* or *GS*), the exception name (if possible), vector offset, stack frame format and a register dump are displayed.

On systems with 68000 core only the most important exceptions can be displayed in detail (because the 68000's stack frame does not contain the vector offset), other exceptions are displayed as "Exception" only.

3.8 I/O Branch Table

MENMON provides a branch table in ROM, with useful routines that can be used by applications or operating system bootstrap loaders. The branch table is located at `0x10880` and has the following format:

Figure 10. *B5-Specific Branch Table Structure*

```
typedef struct {
      void (*put_char) PROTO((char c));      /* output one character */
      int  (*chk_char) PROTO((void));        /* check for input char */
      void (*printf)  PROTO((char *fmt, ... ));/* var. arg printf */
      int (*ee_read)  PROTO((u_int8 *buf, int index, int len)); /* read eeprom */
      int (*ee_write) PROTO((u_int8 *buf, int index, int len)); /* write eeprom*/
      int (*initcons) PROTO((int pollflg));  /* (re-)init console */
      int (*termcons) PROTO((void));         /* deinit console */
      int (*getcons)  PROTO((void));         /* report console 0=ser 1=vga*/
      int (*dummy1) PROTO((void));           /* setled */
      int (*dummy2) PROTO((void));           /* ramsize */
      void (*enter_mm) PROTO((void));        /* enter menmon */
      void (*bfu)PROTO((u_int32 mmg));       /* obsolete */
      int32 (*get_rsr) PROTO((void));        /* read reset status register*/
} B5_iotbl;
#define B5_IOTBL_ADDR (B5_iotbl *)0x10880
#define B5_IOTBL_ADDR_PRM (B5_iotbl *)0x00880 /* for primary menmon */
```

put_char outputs the character <c> unchanged. Output is without interrupts. <CR> is not expanded to <CR><LF>.

chk_char checks whether an input character is present. It returns the character, or -1 if no character is present. *chk_char* can use interrupts (see *initcons*).

printf outputs a formatted string on the display. It expands <CR> (`0xd`) to <CR><LF> (`0xd`,`0xa`).

ee_read copies an area of the serial EEPROM into a buffer of the caller. *index* and *len* must be even, since the EEPROM is 16 bits wide.
Returns: 0 = O.K.
1 = error reading EEPROM

ee_write programs a part of the serial EEPROM with caller's data. *index* and *len* must be even, since the EEPROM is 16 bits wide.
Returns: 0 = O.K.
1 = write error
2 = verify error
3 = erase error

initcons must be executed by an operating bootstrap loader before other actions are carried out at the console. Parameter *pollflg* defines whether input is to be interrupt-driven (0) or polled (1).

Returns: 0 = O.K.
-1 = console can't be set to polling mode

termcons can be called to deactivate the interrupts from the MENMON console.

getcons returns the currently selected console (exists only for compatibility with A8/9/10, always returns 0).

enter_mm can be used by an operating system bootstrap loader to branch back to MENMON.

get_rsr returns MENMON's copy of the MC68360 Reset Status Register

bfu is no longer used.

Notes

- You must be in supervisor state to call these routines.
- Calling conventions are those of the OS-9 Ultra-C compiler:
  First parameter: *d0*, second parameter: *d1*, further parameters: on stack. Return value in d0. For *printf*, this is different. In this case <fmt> is passed in *d0*, all other parameters on stack.

4 Organization of the Board

To install software on the B5 board or to develop low-level software it is essential to be familiar with the board's address and interrupt organization.

4.1 Address Mappings

Address mapping of the B5 microcomputer is basically defined by means of registers on the CPU chip of the MC68360 and in a programmable logic device. The addresses chosen have been optimized for use with the OS-9 operating system.

4.1.1 Local and Global Bus Access by Local CPU

Table 22. *Address Map for Local and Global Bus Access by Local CPU*

Address Range	Size	Function
0x 0000 0000..0000 FFFF	64KB	MEN booter; write-protected
0x 0001 0000..003F FFFF	< 4MB	Global boot Flash
0x 0080 0000..008F FFFF	1MB	M-Module A24/D32
0x 00D0 0000..00DF FFFF	1MB	M-Module A24/D16
0x 00F0 0000..00F0 FFFF	64KB	M-Module A08/D16
0x 00F1 0000..00F1 FFFF	64KB	M-Module IACK
0x 00F8 0000..00F8 FFFF	64KB	B5 Local Control Registers
0x 00F9 0000..00F9 FFFF	64KB	B5 Local Control Registers
0x 00FA 0000..00F7 FFFF	32KB	Real-time clock
0x 00FA 8000..00FA FFFF	32KB	VIC VMEbus control registers
0x 00FD 0000..00FD 7FFF	32KB	CAN Channel B Controller
0x 00FD 8000..00FD FFFF	32KB	CAN Channel A Controller
0x 00FE 0000..00FE FFFF	64KB	B5 Global Control Registers
0x 00FF E000..00FF EFFF	4KB	MC68360 dual-ported RAM
0x 00FF F000..00FF FFFF	4KB	MC68360 internal registers (SIM/CPM)
0x 0200 0000..02FF FFFF	16MB	Global SRAM
0x 0400 0000..04FF FFFF	16MB	Local Flash
0x 0800 0000..083F FFFF	4MB	DRAM, cache inhibit[1]
0x 0840 0000..(depends on configuration)	≤ 32MB	DRAM SIMM, cache inhibit[2]
0x 2800 0000..283F FFFF	4MB	DRAM, cache enabled, copyback[1]
0x 2840 0000..(depends on configuration)	≤ 32MB	DRAM SIMM, cache enabled, copyback[2]
0x 8000 0000..80FF FFFF	16MB	VMEbus A24/D16 BLK Transfer
0x 8800 0000..88FF FFFF	16MB	VMEbus A24/D16
0x 8400 0000..8400 FFFF	64KB	VMEbus A16/D16 BLK Transfer
0x 8C00 0000..8C00 FFFF	64KB	VMEbus A16/D16

[1] Physically identical
[2] Physically identical

4.1.2 Global Bus Access by Another VMEbus Master

The addresses in the following table are **offsets to the A16 base address**. The B5 occupies 4KB in the A16 VMEbus area.

Table 23. *Address Map for Global Bus Access by A16 VMEbus Master*

Address Range	Size	Function
0x 000..7FF	2KB	VIC interprocessor communication
0x 800..9FF	512bytes	M-Module A08/D16
0x A00..BFF	512bytes	VIC VMEbus control registers
0x C00..DFF	512bytes	B5 Global Control Registers
0x E00..FFF	512bytes	M-Module interrupt acknowledge

The addresses in the following table are **offsets to the A24 base address**. The B5 occupies 2MB in the A24 VMEbus area.

Table 24. *Address Map for Global Bus Access by A24 VMEbus Master*

Address Range	Size	Function
0x 00 0000..0F FFFF	1MB	Global SRAM (default)[1]
0x 00 0000..00 FFFF	64KB	MEN booter, write protected[1]
0x 01 0000..0F FFFF	1MB[2]	Boot Flash[1]
0x 10 0000..1F FFFF	1MB	M-Module A24/D16

[1] When programming the boot Flash, Flash is mapped to the SRAM area. The mode can be selected in Global Control Register 1 (see Chapter 4.3.2 Global Control Registers on page 59).

[2] Minus 64KB boot sector.

4.2 Interrupt Handling

The B5's interrupt handler is integrated into the MC68360. There are two groups of interrupts in the MC68360:

- MC68360 SIM System Integration Module (PIT and software watchdog)
- MC68360 CPM Communications Processor Module (all others).

The interrupt levels of the SIM and the CPM can be software-configured.

The VMEbus VIC component also causes interrupts.

The following table gives all interrupt sources indicating their vectors and levels. At system start-up, MENMON performs the following configurations, which are necessary for correct interrupt handling and should not be changed:

Table 25. *Interrupt Priorities and Levels*

Level	Interrupt	Vector
7	ACFAIL# SYSFAIL# Break Point MC68360 Parity Error MC68360 abort button	Autovector 7
6	SIM360: PIT	
5	VMEbus level 6 VMEbus level 5	
4	CPM360: SCC1 SCC2 SCC3 SCC4 SMC1 SMC2 CAN A CAN B	
3	VMEbus level 4 VMEbus level 3	
2	M-Module	Autovector 2
1	Reserved	

4.3 B5 Control Registers

The B5 has some registers for functions that are not implemented using ports or functions of the MC68360. There are local and global registers. Global registers can be accessed by other VMEbus masters via the global bus of the B5.

4.3.1 Local Control Registers

Local Control Register 0 (VMEbus A16/A24 Mask Register, write only, 0x00F80000)

31..24	23..16	15..8	7..0
-	Mask A23..A16	Mask A15..A8	-

Local Control Register 1 (VMEbus A16/A24 Compare Register, write only, 0x00F80004)

31..24	23..16	15..8	7..0
-	Compare A23..A16	Compare A15..A8	-

Local Control Register 2 (Watchdog Mode/Status Register, read/write, 0x00F80008)

31..1	0
-	WRES

Writing any value to this register will start or retrigger the watchdog.

WRES Activity of watchdog reset (read once)
0 = Watchdog reset was not active
1 = Watchdog reset was active

Local Control Register 3 (SIMM ID/Hex Switch, read only, 0x00F90000)

31..12	11..8	7..4	3..0
-	SIMM ID 3..0	Hex Switch	-

If you use SIMM DRAM extensions you can interrogate the SIMM ID from bits 11..8.

The hex switch can be interrogated in this register. Bits 7..4 of this register reflect the switch position. The bits are read as an inverted value: hex switch position 0 corresponds to register value ’F’, position 1 corresponds to ’E’ etc.

Local Control Register 4 (Interrupt 7 Mask Register, read/write, 0x00FE0009)

31..3	2	1	0
-	BPMask	PARMask	ABMask

BPMask MC68360 breakpoint mask
0 = Disable (default)
1 = Enable

PARMask MC68360 parity error mask
0 = Disable (default)
1 = Enable

ABMask Abort button mask
0 = Disable (default)
1 = Enable

Local Control Register 5 (Interrupt 7 Pending Register, read only, 0x00FE000B)

31..3	2	1	0
-	BPPend	PARPend	ABPend

BPPend MC68360 breakpoint interrupt
1 = Pending

PARPend MC68360 parity error interrupt
1 = Pending

ABPend abort button interrupt
1 = Pending

4.3.2 Global Control Registers

The global control registers can be accessed by the CPU and by other VMEbus masters. The addresses given are offsets to the respective base address:

- Access by local CPU: 0x00FE0000
- Access by VMEbus: A16 base address + 0xC00

Global Control Register 0 (Reset Local CPU, read/write, offset 0x01)

15..1	0
-	CPURes

CPURes 0 = Local CPU running (default)
1 = Reset local CPU

Global Control Register 1 (Boot Flash, read/write, offset 0x03)

15..1	0
-	BFlProg

BFlProg 0 = Programming disabled; normal mapping (default)
1 = Programming enabled; Flash mapped on SRAM area

Global Control Register 2 (M-Module IRQ Mode, read/write, offset 0x05)

15..8	7	6	5	4	3	2	1	0
-	IRQ7	IRQ6	IRQ5	IRQ4	IRQ3	IRQ2	IRQ1	IRQHndl

IRQ7..1 0 = VMEbus IRQ7..1 disabled (default)
1 = VMEbus IRQ7..1 enabled

IRQHndl 0 = Local CPU handles M-Module interrupts (default)
1 = VIC handles M-Module interrupts

Global Control Register 3 (M-Module IRQ Mask, read/write, offset 0x07)

15..1	0
-	IRQMask

IRQMask 0 = M-Module interrupt disabled (default)
1 = M-Module interrupt enabled

Global Control Register 4 (LED, read/write, offset 0x0D)

15..4	3	2	1	0
-	LED4	LED3	LED2	LED1

LED4..1 0 = LED off
1 = LED on

Global Control Register 5 (PLD Revision, read only, offset 0x0F)

15..4	3	2	1	0
-	PLD Revision			

Bits 3..0 contain the PLD revision number of B5-41.

4.4 Implementation of MC68360

Table 26. *Chip-Select Signals of MC68360*

Pin	Function
/CS0	Boot EPROM
/CS1	/RAS1 DRAM (SIMM)
/CS2	/RAS2 DRAM (SIMM)
/CS3	/CS SRAM
/CS4	/CS Flash
/CS5	/CS I/O peripherals
/CS6	/RAS6 on-board DRAM
/CS7	Dummy select for dual DMA mode

Table 27. *Use of the MC68360 CPM*

Block	Function
SCC1	Ethernet 1
SCC2	UART2 at SA adapter (P9) or Ethernet 2
SCC3	UART3 at ISO 9141-2 interface
SCC4	UART4 at SA adapter (P10)
SMC1	UART5 at RS232 (with handshake)
SMC2	UART6 at RS232 (without handshake)
IDMA1	Unused
IDMA2	M-Module
SDMA1..14	SCC1..4, SMC1..2, SPI
GPT1	Unused
GPT2	Unused
GPT3	TRIGA M-Module
GPT4	Baud rate generator for UART3

Table 28. *Port Pin Assignment of the MC68360 CPM*

Pin	Signal	Function
Port A		
PA0	RxD1	Ethernet 1 (Receive Data)
PA1	TxD1	Ethernet 1 (Transmit Data)
PA2	RxD2	UART2/Ethernet 2 (Receive Data)
PA3	TxD2	UART2/Ethernet2 (Transmit Data)
PA4	RxD3	ISO 9141-2 K-Line (Receive Data)
PA5	TxD3	ISO 9141-2 K-Line (Transmit Data)
PA6	RxD4	UART4 (Receive Data)
PA7	TxD4	UART4 (Transmit Data)
PA8	CLK1	Ethernet 1 (Transmit CLK)
PA9	CLK2	Ethernet 1 (Receive CLK)
PA10	DSR2/TCLK2	UART2/Ethernet 2 (Transmit CLK)
PA11	RI2/RCLK2	UART2/Ethernet 2 (Receive CLK)
PA13	TOUT3	TRIGA M-Module trigger A
PA14	-	Reserved
PA15	TOUT4	Baud rate generator for UART3
Port B		
PB0	RTS5	UART5 request to send
PB1	DTR2	UART2 data terminal ready
PB2	DTR4	UART4 data terminal ready
PB3	EE_DIN	Write data to EEPROM
PB4	EE_DOUT	Read data from EEPROM
PB5	EE_CS	Select EEPROM
PB6	TxD5	UART5 (transmit data)
PB7	RxD5	UART5 (receive data)
PB8	/DREQ	M-Module DMA request
PB9	/DACK	M-Module DMA acknowledge
PB10	TxD6	UART6 (transmit data)
PB11	RxD6	UART6 (receive data)
PB12	TENA1	Ethernet 1 transmit enable
PB13	TENA2/RTS2	Ethernet 2 transmit enable/UART2 request to send
PB14	RTS3	ISO 9141-2 L-Line
PB15	RTS4	UART4 request to send
PB16	EE_CLK	Clock to EEPROM

Pin	Signal	Function
PB17	ASIZ1	0: VMEbus master A32/user-defined AM code (output pin) 1: VMEbus master A16/A14 AM code (default if pin is input)
Port C		
PC0	REMOTE	State of *Remote2* line at VMEbus
PC1	/REM_ON	Activates the driver for *Remote2* line
PC2	PC2 I/O	TRIGB M-Module trigger B
PC4	CLSN1	Ethernet 1 collision
PC5	RENA1	Ethernet 1 receiver enable
PC6	CTS2/COL2	UART2/Ethernet 2 collision
PC7	DCD2/RENA2	UART2/Ethernet 2 receiver enable
PC8	IRQ_CAN_A	Interrupt request CAN channel A
PC9	IRQ_CAN_B	Interrupt request CAN channel B
PC10	CTS4	UART4 clear to send
PC11	DCD4	UART4 data carrier detect
Port E		
IRQ1,4,6	IPL[2..0]	Interrupt lines to CPU in companion mode
CONFIG0	BCLR0	Unused
IPIPE1	BCLRI	Unused
A[31..28]	A[31..28]	Address lines, cache control
OE	AMUX	Unused
CAS2,3	CAS2,3	DRAM
CAS0,1	CAS0,1	DRAM
AVEC	AVEC	Read-modify-write signaling

5 Appendix

5.1 Literature and WWW Resources

5.1.1 CPU

- M68040 User's Manual, Motorola, 1993
- M68060 User's Manual, Motorola, 1994
- MC68360 User's Manual, 1993

www.motorola.com

5.1.2 Ethernet

- Ethernet in general:
  - The Ethernet, A Local Area Network, Data Link Layer and Physical Layer Specifications, Version 2.0; 1982; Digital Equipment Corporation, Intel Corp., Xerox Corp.
  - ANSI/IEEE 802.3-1996, Information Technology - Telecommunications and Information Exchange between Systems - Local and Metropolitan Area Networks - Specific Requirements - Part 3: Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications; 1996; IEEE
    www.ieee.org
- www.ethermanage.com/ethernet/
  links to documents describing Ethernet, components, media, the Auto-Negotiation system, multi-segment configuration guidelines, and information on the Ethernet Configuration Guidelines book
- www.iol.unh.edu/training/ethernet.html
  collection of links to Ethernet information, including tutorials, FAQs, and guides
- www.made-it.com/CKP/ieee8023.html
  Connectivity Knowledge Platform at Made IT technology information service, with lots of general information on Ethernet

5.1.3 CAN Bus

- www.can-cia.de/
  CAN in Automation e. V.
- 141.44.61.248/NT/CAN/Welcome.html
  CAN Home Page, University of Magdeburg
- www.hitex.co.uk/CAN/canarticle.html

5.1.4 VMEbus

- VMEbus chip VIC068A:
  Applications Handbook; April 1994
  Cypress Semiconductor, 3901 North First Street, San Jose, CA 95134
  www.cypress.com
- VMEbus General:
  - The VMEbus Specification, 1989
  - The VMEbus Handbook, Wade D.Peterson, 1989

  VMEbus International Trade Association
  www.vita.com

5.1.5 M-Modules

- M-Module Standard:
  ANSI/VITA 12-1996, M-Module Specification;
  VMEbus International Trade Association
  www.vita.com

5.2 Board Revisions

Table 29. *Table of Hardware Revisions*

Revision	Comment	Restrictions
00.xx	Prototype	None known
01.xx	First revision	None known
02.xx	Second revision	None known
03.xx	Third revision	None known
04.xx	Fourth revision	None known

5.3 Component Plans

Figure 11. *Component Plan of B5 Hardware Revision 04 — top side*

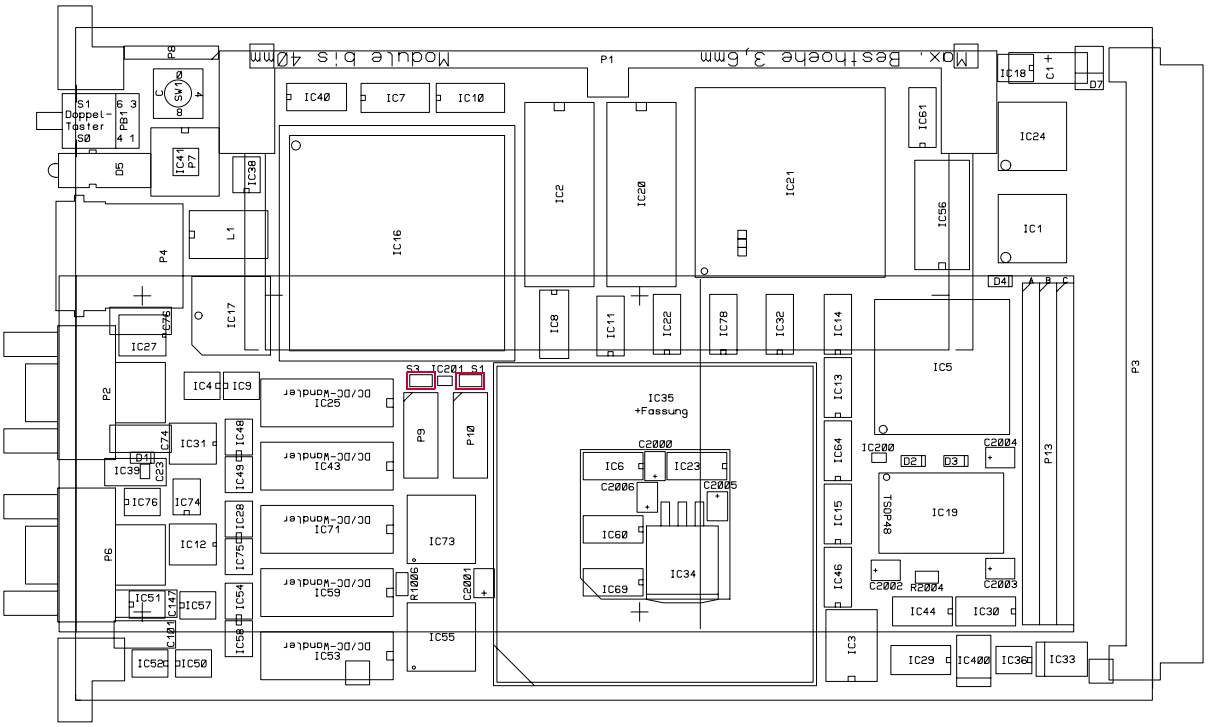
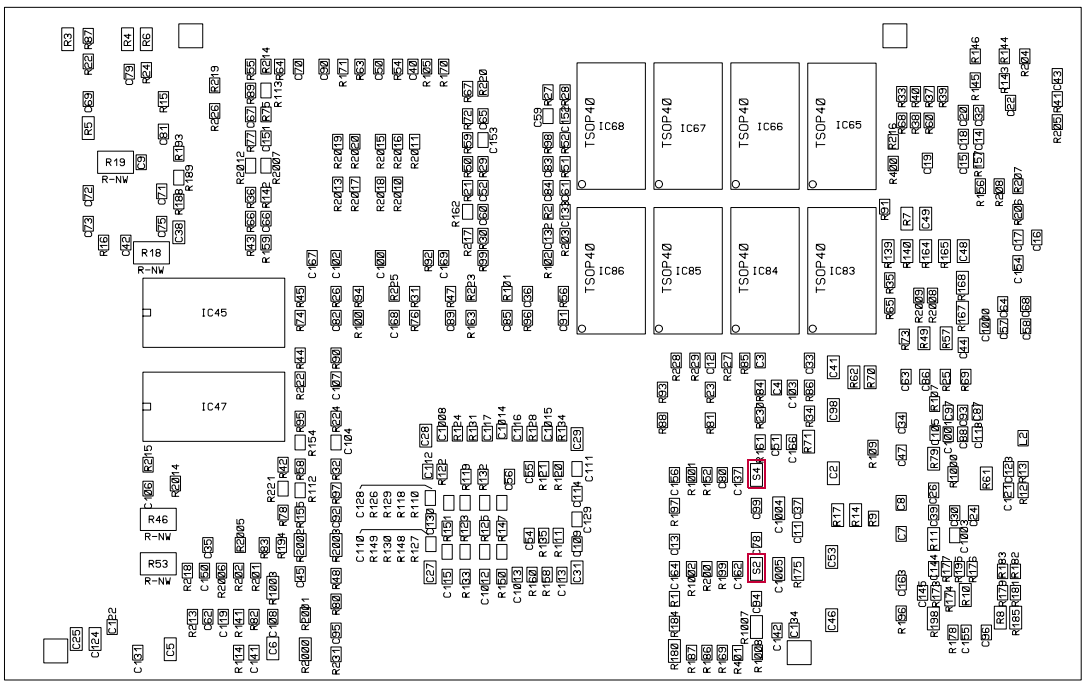


Figure 12. *Component Plan of B5 Hardware Revision 04 — bottom side*



You can request the circuit diagrams for the current revision of the product described in this manual by completely filling out and signing the following non-disclosure agreement.

Please send the agreement to MEN by mail. We will send you the circuit diagrams along with a copy of the completely signed agreement by return mail.

MEN reserves the right to refuse sending of confidential information for any reason that MEN may consider substantial.

mikro elektronik
gmbh · nürnberg

Non-Disclosure Agreement

for Circuit Diagrams provided by MEN Mikro Elektronik GmbH

between

MEN Mikro Elektronik GmbH
Neuwieder Straße 7
D-90411 Nürnberg

("**MEN**")

and

_____
_____
_____
_____

("**Recipient**")

We confirm the following Agreement:

MEN		**Recipient**	
Date:	_____	Date:	_____
Name:	_____	Name:	_____
Function:	_____	Function:	_____
Signature:		Signature:	
_____		_____	

The following Agreement is valid as of the date of MEN's signature.

MEN Mikro Elektronik GmbH

Neuwieder Straße 7
90411 Nürnberg
Deutschland

Tel. +49-911-99 33 5-0
Fax +49-911-99 33 5-99

E-Mail info@men.de
www.men.de

Geschäftsführer Manfred Schmitz, Udo Fuchs Handelsregister Nürnberg HRB 5540 UST-ID-Nr. DE 133 528 744

Deutsche Bank AG Kto. Nr. 0390 211, BLZ 760 700 12 HypoVereinsbank Kto. Nr. 1560 224 300, BLZ 760 200 70

ISO 9001 zertifiziert

1 Subject

The subject of this Agreement is to protect all information contained in the circuit diagrams of the following product:

Article Number: _____ [filled out by recipient]

MEN provides the recipient with the circuit diagrams requested through this Agreement only for information.

2 Responsibilities of MEN

Information in the circuit diagrams has been carefully checked and is believed to be accurate as of the date of release; however, no responsibility is assumed for inaccuracies. MEN will not be liable for any consequential or incidental damages arising from reliance on the accuracy of the circuit diagrams. The information contained therein is subject to change without notice.

3 Responsibilities of Recipient

The recipient, obtaining confidential information from MEN because of this Agreement, is obliged to protect this information.

The recipient will not pass on the circuit diagrams or parts thereof to third parties, neither to individuals nor to companies or other organizations, without the written permission by MEN. The circuit diagrams may only be passed to employees who need to know their content. The recipient protects the confidential information obtained through the circuit diagrams in the same way as he protects his own confidential information of the same kind.

4 Violation of Agreement

The recipient is liable for any damage arising from violation of one or several sections of this Agreement. MEN has a right to claim damages amounting to the damage caused, at least to €100,000.

5 Other Agreements

MEN reserves the right to pass on its circuit diagrams to other business relations to the extent permitted by the Agreement.

Neither MEN nor the recipient acquire licenses for the right of intellectual possession of the other party because of this Agreement.

This Agreement does not result in any obligation of the parties to purchase services or products from the other party.

6 Validity of Agreement

The period after which MEN agrees not to assert claims against the recipient with respect to the confidential information disclosed under this Agreement shall be _____ months [filled out by MEN]. (Not less than twenty-four (24) nor more than sixty (60) months.)

7 General

If any provision of this Agreement is held to be invalid, such decision shall not affect the validity of the remaining provisions and such provision shall be reformed to and only to the extent necessary to make it effective and legal.

This Agreement is only effective if signed by both parties.

Amendments to this Agreement can be adopted only in writing. There are no supplementary oral agreements.

This Agreement shall be governed by German Law.

The court of jurisdiction shall be Nuremberg.

MEN Mikro Elektronik GmbH

Neuwieder Straße 7
90411 Nürnberg
Deutschland

Tel. +49-911-99 33 5-0
Fax +49-911-99 33 5-99

E-Mail info@men.de
www.men.de

Geschäftsführer Manfred Schmitz, Udo Fuchs Handelsregister Nürnberg HRB 5540 UST-ID-Nr. DE 133 528 744

Deutsche Bank AG Kto. Nr. 0390 211, BLZ 760 700 12 HypoVereinsbank Kto. Nr. 1560 224 300, BLZ 760 200 70

ISO 9001 zertifiziert